NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

K86776

METHODOLOGIES FOR RESOLVING ANOMALOUS POSITION INFORMATION IN TORPEDO RANGE TRACKING USING SIMULATION

by

William M. Kroshl

March 1988

Thesis Advisor: R. R. Read

Approved for public release; distribution is unlimited

T239047

URITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; Distribution is unlimited
b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) 55	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO

1. TITLE (Include Security Classification)
METHODOLOGIES FOR RESOLVING ANOMALOUS POSITION NFORMATION IN TORPEDO RANGE TRACKING USING SIMULATION

2. PERSONAL AUTHOR(S)
KROSHL, WILLIAM M.

13a. TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM _____ TO _____	1988 March	163

6 SUPPLEMENTARY NOTATION "The views expressed in this thesis are those of the author nd do not reflect the official policy or position of the Department of efense or the U.S. Government."

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Underwater Acoustics, Raytracing, Simulation, Torpedo, Underwater Range

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

This thesis investigates the problem of resolving dual path records of torpedoes being tracked on an acoustic range using short baseline arrays. An acoustic signal is sent out by a torpedo at short intervals. This signal is then received by the four hydrophones of a short baseline array. Arrival time differences in the signal are used to determine an estimated position for the torpedo at the time the signal was emitted using spherical equations and acoustic raytracing. In those areas where two arrays can track a target simultaneously, two sets of estimated positions are generated. These estimates usually do not coincide.

A simulation of the range is developed using actual range positions and sound velocity data. Deliberate errors are then range introduced into the sound velocity profile data and the timing data. Three methods of resolving the resulting positional ambiguity are presented. Each

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Prof. R. R. Read	22b TELEPHONE (Include Area Code) (408) 646-2382 22c OFFICE SYMBOL 55Re

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office: 1986—606-24.

method is compared to the actual position for idealized (no deliberate error) and the deliberate error models.

Approved for public release; distribution is unlimited.

Methodologies for Resolving Anomalous Position Information in Torpedo Range Tracking Using Simulation

by

William M. Kroshl
Lieutenant Commander, United States Navy
B.A., Northwestern University, 1975

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1988

R3676
C.1

ABSTRACT

This thesis investigates the problem of resolving dual path records of torpedoes being tracked on an acoustic range using short baseline arrays. An acoustic signal is sent out by a torpedo at short intervals. This signal is then received by the four hydrophones of a short baseline array. Arrival time differences in the signal are used to determine an estimated position for the torpedo at the time the signal was emitted using spherical equations and acoustic raytracing. In those areas where two arrays can track a target simultaneously, two sets of estimated positions are generated. These estimates usually do not coincide.

A simulation of the range is developed using actual range positions and sound velocity data. Deliberate errors are then introduced into the sound velocity profile data and the timing data. Three methods of resolving the resulting positional ambiguity are presented. Each method is compared to the actual position for idealized (no deliberate error) and the deliberate error models.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

I. INTRODUCTION AND BACKGROUND 1

A. INTRODUCTION 1

B. PURPOSE 2

C. RANGE DESCRIPTION AND OPERATION 4

D. RANGE POSITION DETERMINATION 8

II. RAYTRACING 10

A. BACKGROUND 10

B. ISOGRADIENT RAYTRACING 13

C. MULTILAYER RAYTRACING 16

III. GENERAL RANGE SIMULATION MODEL 20

A. BACKGROUND 20

B. ANGLE AND TIME GENERATION ALGORITHM 20

C. ACCURACY OF ALGORITHM 26

D. NANOOSE SIMULATION 28

E. MODEL SPECIFICS 29

IV. RANGE ERROR MODELS 32

A. OVERVIEW 32

B. NAVY BASE CASE 32

C. CONSTANT TIME ERRORS 34

D. RANDOM TIME ERRORS 35

E. CONSTANT VELOCITY ERRORS 36

F. RANDOM VELOCITY ERRORS 37

V. POSITION CORRECTION MODELS 49

A. INTERSECTION METHOD 49

B. MIDPOINT METHOD 50

C. DELTA METHOD 50

VI. CONCLUSIONS AND RECOMMENDATIONS 66

A. GENERAL 66

B. RANGE OPERATIONS AND ERROR MECHANISMS 66

C. INTERSECTION METHOD 68

D. MIDPOINT METHOD 68

E. DELTA METHOD 68

F. RECOMMENDATIONS FOR FURTHER RESEARCH 69

APPENDIX A: COMPUTER PROGRAM FOR ISOGRADIENT RAYTRACING 70

APPENDIX B: COMPUTER PROGRAM FOR TIME AND ELEVATION ANGLE GENERATION 74

APPENDIX C: SIMULATION PROGRAMS 77

APPENDIX D: SAMPLE DEPTH VELOCITY PROFILE 90

APPENDIX E: NANOOSE RANGE SENSOR POSITIONS 92

APPENDIX F: CASE I MODEL GRAPHS 93

APPENDIX G: CASE II MODEL GRAPHS 99

APPENDIX H: CASE III MODEL GRAPHS 105

APPENDIX I: CASE IV MODEL GRAPHS 111

APPENDIX J: CASE V MODEL GRAPHS 117

APPENDIX K: CASE VI MODEL GRAPHS 123

APPENDIX L: CASE VI MODEL GRAPHS 129

APPENDIX M: CASE VII MODEL GRAPHS 135

APPENDIX N: SIMULATION LOGICAL FLOW PATH 141

LIST OF REFERENCES 149

INITIAL DISTRIBUTION LIST 150

LIST OF TABLES

IV.1	COMPARISON ERRORS	38
IV.2	CONSTANT TIME ERROR (+0.0005)	39
IV.3	CONSTANT TIME ERROR (-0.0005)	40
IV.4	NORMAL TIME ERRORS (INDEPENDENT)	41
IV.5	NORMAL TIME ERRORS	42
IV.6	CONSTANT VELOCITY ERRORS (+0.005)	43
IV.7	CONSTANT VELOCITY ERRORS (+0.05)	44
IV.8	NORMAL VELOCITY ERRORS	45
V.1	INTERSECTION METHOD	54
V.2	MIDPOINT METHOD	56
V.3	DELTA METHOD (CASE I)	58
V.4	DELTA METHOD (CASE II)	59
V.5	DELTA METHOD (CASE III)	60
V.6	DELTA METHOD (CASE IV)	61
V.7	DELTA METHOD (CASE V)	62
V.8	DELTA METHOD (CASE VI)	63
V.9	DELTA METHOD (CASE VII)	64
V.10	DELTA METHOD (CASE VIII)	65

LIST OF FIGURES

2.1 ISOGRADIENT RAYTRACING 14

2.2 MULTILAYER RANGES 18

3.1 MULTILAYER GEOMETRY 23

3.2 SIMULATION ACCURACY 27

3.3 SIMULATION TRACKS 31

4.1 SAMPLE TRACKS: CASE II 46

4.2 SAMPLE TRACKS: CASE III 47

4.3 SAMPLE TRACKS: CASE IV 48

ACKNOWLEDGEMENTS

For her support and encouragement, I dedicate this thesis to my wife Tina.

I. INTRODUCTION AND BACKGROUND

A. INTRODUCTION

Torpedoes are tested for proper operation at the Dabob Bay and Nanoose underwater ranges operated by the Naval Undersea Warfare Engineering Station, Keyport, WA. (NUWES). Data for this research was taken from the Nanoose range.

On the ranges, the torpedo is tracked by a system of hydrophone arrays on the ocean floor. Due to acoustic attenuation, each array has a finite area of coverage. As the torpedo moves down the range, it moves from the coverage of one array to another. In general, the arrays are located to minimize overlap of coverage, but some does occur between adjacent arrays. This area is called the "crossover zone".

The primary purpose of the ranges is to test the torpedoes for proper operation. In order to determine if the torpedo is acquiring, maintaining or reacquiring, and homing on its target properly, accurate tracking of the torpedo is essential.

However, the data provided by the range is rarely "clean". The data quality varies markedly. This is particularly evident in the crossover zone. The estimated tracks generated by each of the arrays rarely coincide. There are several possible sources for this noise in the

data. These sources may be operating separately or in concert.

If the tracking were perfect, but the actual positions of the arrays were different from the ones plotted, errors in position would occur.

A second possible source of error is that the system is very sensitive to errors in timing. Any error in timing will have a direct effect on the accuracy of the position generated.

A third source of error is due to the water conditions. The raytracing procedure depends on the velocity of sound at each depth of water. A measurement of the sound velocity, from the surface to the depth of the deepest array, is taken prior to each day's operation. This information, called a sound velocity profile (SVP), is then assumed to be constant throughout the range for that day's operations. The velocity of sound in water is highly variable, and influenced by depth, temperature, and salinity changes. Any errors, either a bias of the measurement or inhomogeneity from one part of the range to another, will affect accuracy of the range.

B. PURPOSE

Professor Robert R. Read, U.S. Naval Postgraduate School, has been working on a method of using the crossover data to calibrate the array position data, and has developed the KEYMAIN program for this purpose.

In the course of this work, NUWES determined that an accurate, realistic simulation of the operation of the range was needed to assist in understanding the error mechanisms encountered in range operation. Once the simulation was operational, various methodologies of estimating a true position from the "noisy" data could be developed and tested.

Simulation was considered to be the best method for exploration of error mechanisms for several reasons. The most important reason is that there is no way of directly and independently monitoring the performance of the range. The only way to determine the position of a torpedo on the range is to use the range itself. Thus, there is no "correct" position for the torpedo at any time, only estimated positions provided by each array.

Simulation allows the sources of "noise" in the data to be studied separately and in concert. Methods to arrive at a more accurate estimate of the true position can be tested under different conditions, and then compared against an actual position. This is not possible in actual range experiments.

Earlier simulations of the range did not use the actual range raytracing procedures and methods, and did not allow actual range operations to be analyzed. Main [Ref. 1], for example, used a simulation to study the effects of various position determination methods. However, these earlier

models did not allow testing of various error mechanisms (noise sources) and correction procedures.

This thesis presents the author's simulation of the range operation. Models of the error mechanism for timing and sound velocity are presented. Three methods of resolving the position ambiguity in the crossover zone are presented, and then tested against the simulation. The question of positional error of the arrays will not be addressed.

C. RANGE DESCRIPTION AND OPERATION

The range consists of a number of short baseline hydrophone arrays, anchored on the ocean floor. The arrays are at depths ranging from 1000 to 1300 feet, and spaced approximately 5000 feet apart.

Each array consists of a system of four hydrophones, arranged at the corners of a cube. If you consider the acoustic center of the array as (0,0,0), and D as the distance between the hydrophones, the coordinates of the hydrophones in the local coordinate system are as follows:

$$\begin{aligned} x: &\quad (D,-D,-D)/2 \\ y: &\quad (-D,D,-D)/2 \\ z: &\quad (-D,-D,D)/2 \\ c: &\quad (-D,-D,-D)/2 \end{aligned} \tag{1.1}$$

For the range simulated in this study, the value of D was a constant 30 feet. Each of these arrays is linked with an ashore computer by buried cable. A pinger located

in the torpedo sends out a short acoustic pulse at regular intervals. A clock on the torpedo is synchronized with the time standard on the range. The time of receipt of the sound pulse (ping) is recorded at each of the four hydrophones. This allows the calculations of the time of flight of the sound to each of the four hydrophones (X, Y, Z, and C).

The first step in calculation of the position is to obtain an approximation for the actual position. This apparent position is calculated by assuming that the speed of sound is constant throughout the path of the sound. This position is then refined by raytracing.

If the speed of sound is constant, the wave front of sound from the pinger will form an expanding sphere. Knowing the time of travel and the velocity of sound, the position can be calculated. The constant velocity used is the speed of sound at the hydrophone.

All coordinates are expressed in the local system, based in the acoustic center of the array as (0, 0, 0). Let (X, Y, Z) be the assumed position to be calculated. If T_i is the time of travel to hydrophone i, where i is (x,y,z, or c), and V is the velocity of sound at thathydrophone, then the distance between the sound source and hydrophone can be expressed as:

$$dist = (V)(T_i) \qquad (1.2)$$

The distance can also be expressed as the euclidian norm between the hydrophone and the target. Combining these two methods of expressing distance yields the following system of equations:

$$\begin{aligned}
(X - D/2)^2 + (Y + D/2)^2 + (Z + D/2)^2 &= V^2T^2_x \\
(X + D/2)^2 + (Y - D/2)^2 + (Z + D/2)^2 &= V^2T^2_y \\
(X + D/2)^2 + (Y + D/2)^2 + (Z - D/2)^2 &= V^2T^2_z \\
(X + D/2)^2 + (Y + D/2)^2 + (Z + D/2)^2 &= V^2T^2_c
\end{aligned} \tag{1.3}$$

If the times and velocities are known, this forms a system of four equations in three unknowns. Generally, such as system will not have any exact solution. Even if the times recorded were exact, the distance equations using velocity correspond to the actual, curved ray paths, while the equations using distance refer to the straight line distance.

Rather than discard one of the equations, and the information that it contains, the method used on the range (hereinafter called the NAVY method), involves the following steps. First, the last equation is subtracted from each of the other three equations. This yields the following system of equations:

$$\begin{aligned}
(X - D/2)^2 - (X + D/2)^2 &= V^2(T^2_c - T^2_x) \\
(Y - D/2)^2 - (Y + D/2)^2 &= V^2(T^2_c - T^2_y) \\
(Z - D/2)^2 - (Z + D/2)^2 &= V^2(T^2_c - T^2_z)
\end{aligned} \tag{1.4}$$

Solving this system of equations yields the following values for the apparent position of the torpedo:

$$X = V^2 (T_c^2 - T_x^2)/2D \qquad (1.5)$$
$$Y = V^2 (T_c^2 - T_y^2)/2D$$
$$Z = V^2 (T_c^2 - T_z^2)/2D$$

This initial position (X, Y, Z) is then refined by raytracing. The raytracing procedure will be treated in detail in the next chapter. This procedure requires the initial angle of elevation and the time of travel of the sound ray from the torpedo to the acoustic center of the array. The initial angle is estimated using the values obtained in Equation 1.5, above, as follows:

$$A = \arcsin(Z / (X^2 + Y^2 + Z^2)^{.5}) \qquad (1.6)$$

Let R be the horizontal range from the torpedo to the acoustic center of the range, and R_c be the horizontal range from the c hydrophone of the array to the torpedo. R and R_c are then used to define the estimated time of travel to the acoustic center of the system using the following relationships:

$$R = (X^2 + Y^2 + Z^2)^{.5} \qquad (1.7)$$
$$R_c = ((X+D/2)^2 + (Y+D/2)^2 + (Z+D/2)^2)^{.5} \qquad (1.8)$$

$$T = (T_c)\ (R)\ /R_c \qquad (1.9)$$

Main [Ref. 1] gives another description of the range operation and the derivation of these equations.

D. RANGE POSITION DETERMINATION

The result of the initial position determination operation is a position for the torpedo in the local coordinate system (X, Y, Z), and the assumed angle and time of travel at the acoustic center of the system (A, T). The initial position estimate is refined by the raytracing procedure to yield a final position of the torpedo. This sequence is repeated for each ping of the sound source, yielding a track for the torpedo in the local coordinate system of the array.

The position in local coordinates must be transformed to the range coordinate system. There are two general types of corrections to be applied: translational and rotational.

The translational corrections are applied in an additive manner, using the positions of the array's acoustic center expressed in the range coordinate system, which measures position as downrange, crossrange, and depth as measured positively downward from the water surface.

The rotational correction is expressed in terms of Euler Angles, and account for the "twisting" of the individual array from the range alignment. These angles,

called X-Tilt, Y-Tilt, and Z-Rot, are all measured conventionally from their respective positive axis.

The result is a position for the torpedo as measured by one array, expressed in global range coordinates and standard depth convention. This procedure is then repeated for each ping, building up a track of the pinger's (and hence the torpedo's) position. In those regions where two arrays can receive the sound signal (crossover zone), two tracks are generated, one from each array.

II. RAYTRACING

A. BACKGROUND

Raytracing is used to refine the approximate position obtained by the NAVY method described previously. It involves tracing out the actual path of sound from the source to the sensor. The key assumption for raytracing is that the ray paths are everywhere normal to their own wave front. While this assumption is guaranteed for spherically expanding waves, such as those from a point source, it can be guaranteed if several other conditions are met. These are as follows:

(1) if the path of the sound wave is curving due to changes in the index of refraction of the medium, the radius of curvature must be large relative to the wavelength of the sound; (2) the acoustic index of refraction should not change appreciably over the wavelength; (3) the percentage change of the amplitude of the signal over a wavelength should be small. For a complete discussion of raytracing, a standard reference text on acoustics, such as Camp [Ref. 2], is recommended.

The path of sound underwater is curved due to changes in the velocity of sound. In general, velocity tends to increase with depth. The temperature of the water has a great effect, with the sound velocity increasing in colder

water. Salinity can have a great effect on velocity. This effect is extremely important in areas where fresh or brackish water is mixing with salt water, such as bays and river mouths.

There are two general models used in raytracing where the speed of sound is changing with depth. Both methods involve dividing the water into relatively thin (25 feet, for this study) layers, and then modeling the ray path through each successive layer.

The first method is ISOSPEED. In the isospeed model, the speed of sound is assumed to be constant in each layer. The velocity chosen is the mean velocity for the layer. By repeated application of Snell's law, the ray path is described from layer to layer.

The second method is ISOGRADIENT. The velocity of sound in water generally increases with depth. Let V_z represent the velocity of sound in water at depth z. In this model, the velocity of sound at a depth Z is modeled by the following equation:

$$V_z = V_0 + (V_1)\ (Z) \qquad (2.1)$$

V_0 is the velocity at the waters surface, and V_1, called the gradient, represents the change in velocity as the depth Z increases. If using a model with a single layer, V_0 and V_1 are determined by a least squares regression of velocity and depth. In the multilayer model, V_0 is the

velocity at the top of each layer, and V_1 is the gradient within that layer.

In simulating the operation of the range, it was necessary to be able to calculate the times (and hence the elevation angles) at each of the four hydrophones (X, Y, Z, and C) given the positions of the hydrophone and the torpedo. The isogradient and the isospeed approach both usually yield positions that are within .25 feet of each other when used to raytrace from a given point, using the same elevation angle, time, and sound velocity profile.

These methods all involve raytracing from a given point, for a given amount of time, at a given angle, using a specific sound velocity profile. For the simulation, using a specific sound velocity profile, the time and angle had to be determined for two given positions. Testing both approaches proved this to be an extremely delicate calculation. An algorithm using the isospeed method was less analytically complex and computationally less intensive, but less accurate than the isogradient case. The positions generated using the isospeed model were within 2-3 feet of the actual positions when raytracing back using the generated time and elevation angle. Therefore, isospeed raytracing was dropped from consideration. The rest of the this paper will involve only isogradient raytracing.

B. ISOGRADIENT RAYTRACING

To begin a discussion of isogradient raytracing, assume that we are dealing with only one layer of water, and that the velocity of sound in that layer is given by Equation 2.1. Establish a coordinate system in the vertical plane that contains the hydrophone (S_1, S_2) and the sound source (P_1, P_2), where the first number of the ordered pair refers to the horizontal range measured from a vertical starting axis, and the second number refers to the depth, measured positively in a downward direction from the water surface (see Figure 2.1).

Under these conditions, the ray path is a circular arc with center (C_1, C_2). C_2, the z co-ordinate of the circle center, is given by equation 2.2, following:

$$C_2 = -V_0/V_1 \tag{2.2}$$

Let R be the length of the radius of the circle, ie the length of the line segments joining (C_1, C_2) with both the sensor location (S_1, S_2), and the sound source (P_1, P_2). Calculating the distance yields the following two equations:

$$(S_1 - C_1)^2 + (S_2 - C_2)^2 = R^2 \tag{2.3}$$
$$(P_1 - C_1)^2 + (P_2 - C_2)^2 = R^2$$

Combining these two equations to solve for C_1 yields the following:

$$C_1 = \frac{(P_1 + S_1)}{2} + \frac{(P_2 - S_2)}{2(P_1 - S_1)}(P_2 + S_2 - 2C_2) \tag{2.4}$$

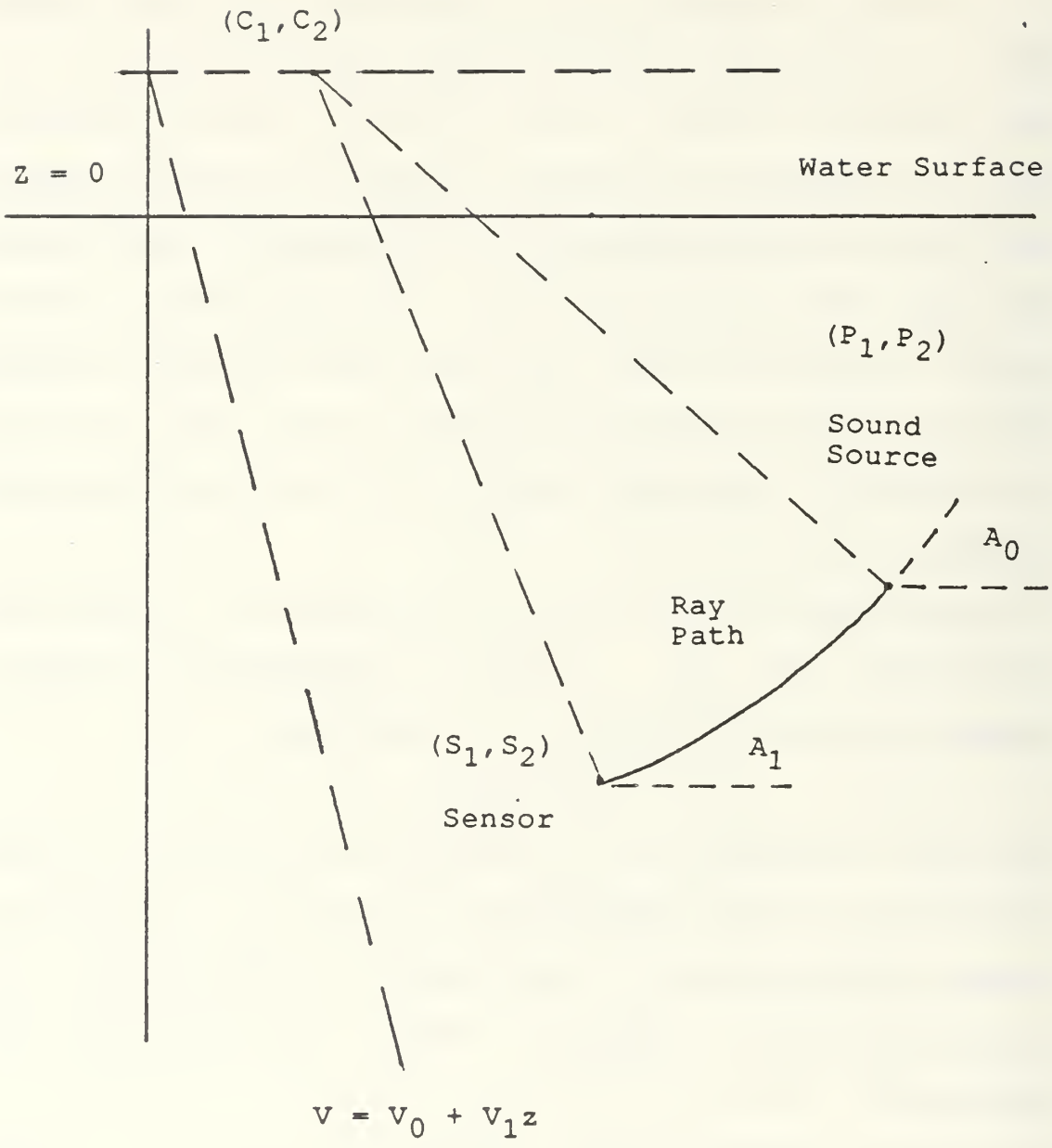


Figure 2.1 Isogradient Raytracing

Using trigonometric substitution, C_1 can also be expressed in the following manner:

$$C_1 = (C_2 - S_2) \tan(A) \qquad (2.5)$$

where A is the elevation angle at the sensor, measured from the horizontal.

The equation for the length of the line segment R can now be obtained as follows:

$$R = ((S_1 - C_1)^2 + (S_2 - C_2)^2)^{.5} \qquad (2.6)$$

The sound velocity does not remain constant along this arc. Therefore, the velocity must be integrated with respect to distance along the ray path to determine the time.

If we let A_0 be the elevation angle of the ray at the sound source, and A_1 be the elevation angle of the ray at the hydrophone, then the time of transit T is represented by the following equation:

$$T = \frac{1}{V_1} \ln \left[\frac{\cos(A_1)}{\cos(A_0)} \frac{(1 + \sin(A_0))}{(1 + \sin(A_1))} \right] \qquad (2.7)$$

For a complete discussion of the derivation of these equations, see Camp [Ref. 2] or Coppens [Ref. 3].

The time can also be expressed in the following form by using substitution of known quantities:

$$T = \frac{1}{V_1} \ln \left[\frac{S_2 - C_2}{P_2 - C_2} \frac{(R + P_1 - C_1)}{(R + S_1 - C_1)} \right] \qquad (2.8)$$

Other quantities needed can be easily obtained by trigonometric substitution into these equations.

C. MULTILAYER RAYTRACING

In actuality, the isogradient approach is an approximation because the gradient is not linear. Due to the accuracies required on the torpedo tracking range, a more accurate method is needed. This is obtained by dividing the water into 25 foot thick layers, starting at the ocean surface and going down to the ocean floor. The speed of sound is measured every 25 feet. The difference in sound velocity from one measurement to another is used to determine a velocity gradient for each layer. The isogradient approach is then used within each layer.

The inputs to the procedure are the sound velocity profile, which gives the velocity of sound at depth increments of twenty five feet, the time of transit of the sound from the torpedo to the acoustic center of the array, and the initial elevation angle of the ray. These are obtained from the four hydrophone transit times as outlined in Chapter 1.

Coppens [Ref. 3] discusses the concept of the Ray Invariant, RV. The ray invariant is a function of the elevation angle at the array (A) and the velocity of sound at the hydrophone (V):

$$RV = \cos(A) / V \qquad (2.9)$$

Let Z equal the difference between the array depth and the source depth, and Z_j be the depth component of the

j th layer. Note that Z_j will be 25 feet for all j except for the layers which contain the array center and the source. In the same manner, let R equal the horizontal range between the hydrophone and the source, and R_j be the range component within the j th layer (see Figure 2.2).

Therefore, by summing over all j between array center and source,

$$Z = \Sigma_j Z_j \qquad (2.10)$$

$$R = \Sigma_j R_j$$

The raytracing algorithm begins by calculating the center of the ray path C, based on the initial angle A and the ray invariant. The time that is used to traverse the depth from the array center to the top of the layer containing the array center is calculated, along with the range. The exit angle of the ray at the top of the layer is calculated. The total transit time is decremented by the layer transit time, and the depth of the hydrophone is decremented by Z_s.

The exit angle is then used as the initial elevation angle for the new layer (s-1). Using the velocity at the top of the new layer, and the layer gradient, the process is then repeated. This continues until the last layer.

The result of this is depth of the sound source, and the horizontal range of the source from the acoustic center of the array, all in the vertical plane containing the hydrophone and the source. This determines a final

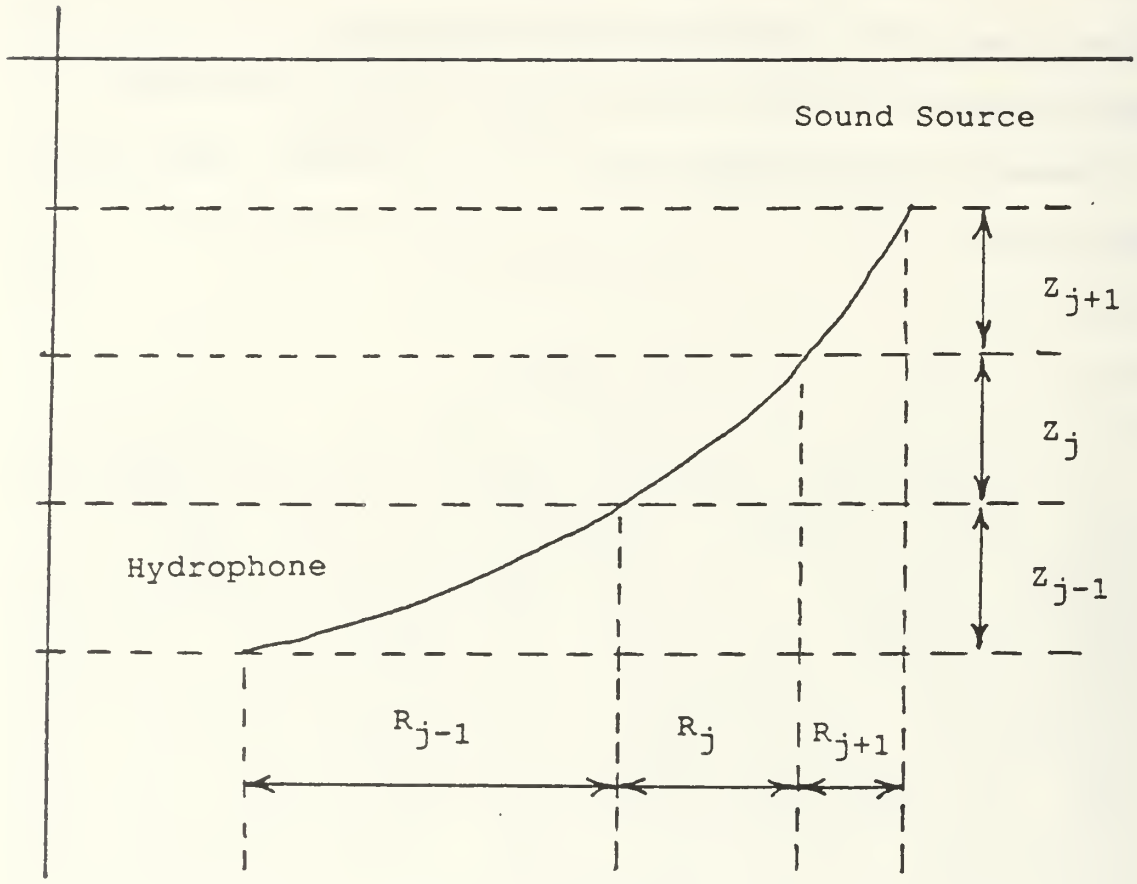


Figure 2.2 Multilayer Ranges

position in local coordinates. This position is converted to range coordinates for the final position, as indicated by one array, for the location of the torpedo at one instant in time (one point count). This procedure is repeated for all arrays in range at each point count to build up a track for the torpedo.

The author's FORTRAN 77 program to implement this procedure is presented in Appendix A. The program uses double precision arithmetic throughout.

III. GENERAL RANGE SIMUALTION MODEL

A. BACKGROUND

In order to accurately simulate the range, the angle and time of the sound ray from a known sound source position to a hydrophone of known position, using a known sound velocity profile, had to be calculated.

As discussed in the last chapter, only the isogradient method provided the accuracy needed for the range simulation. While a closed form solution can be obtained for the time and angle in a single layer example, this cannot be done for the multilayer case. However, this can be accomplished with an iterative technique. This chapter outlines the algorithm for angle and time generation, and presents a numerical example as an accuracy check. The FORTRAN 77 code for this method is contained in Appendix B. Finally, the simulation of the Nanoose range is presented.

B. ANGLE AND TIME GENERATION ALGORITHM

When raytracing, you start with a given angle and trace out a path until transit time is exhausted. The stopping point determines the depth and range of the sound source. In the time and angle generation algorithm, the actual range of the source from the hydrophone is known, as are the depths of the hydrophone and the source. The basic procedure is to start at the depth of the hydrophone, at an

assumed elevation angle, and then start tracing until reaching the depth of the sound source. Then compare the range to the actual range, adjust the angle, and try again. Thus, the elevation angle at the hydrophone is varied until the range coincides with the actual range. The time calculated is the time to travel from the depth of the sensor to the depth of the sound source, at the initial angle given.

The subscript k refers to the layer, counting the layer which ends at the water surface as layer 1. Layer j is the layer containing the sensor, and layer i is the layer containing the sound source. All ranges are horizontal ranges, and all depths are measured positively downward from the water surface. These distances are measured in a vertical plane containing the sensor and the sound source.

Equations 2.9 and 2.10 form the basis for this method. The range R is considered to be the sum of the individual ranges in each layer, R_k. The depth Z is the sum of the individual depth of each layer Z_k. The range R can be expressed as a function of the range in each layer k in the form

$$R = \sum_{k=i}^{j} R_k \qquad (3.1)$$

The key to this process is the ray invariant, defined in Equation 2.9. The ray invariant relates the elevation

angle at the start of each layer to the initial angle at the hydrophone. If the velocity of sound at a given depth is known, the ray invariant can be used to determine the initial angle at each layer k, measuring the angle at the point where the sound ray intersects the lower boundary of the layer. If RV is defined as the ray invariant and V_k is velocity of sound at that level, then A_k, the initial angle at layer k, is defined by:

$$A_k = \arccos (RV\ V_k) \qquad (3.2)$$

Also, note that the exit angle of layer k, is the angle at the upper boundary of layer k, is the same as the initial angle of layer k-1.

Now define $C1_k$ and $C2_k$ as the downrange and depth coordinates of the center of the circle used to represent the isogradient ray path in layer k. These distances are measured in a vertical plane containing the hydrophone and the sound source. Let S1 and S2 define the hydrophone coordinates (downrange and depth), T_k the time to transit layer k, L_k the depth of the midpoint of each layer, G_k as the sound velocity gradient within layer k, and Z_k the "thickness" of each layer (see Figure 3.1).

The first step is to redefine boundaries of the uppermost layer to coincide with the depth of the sound source, and the deepest layer to coincide with the depth of the hydrophone.

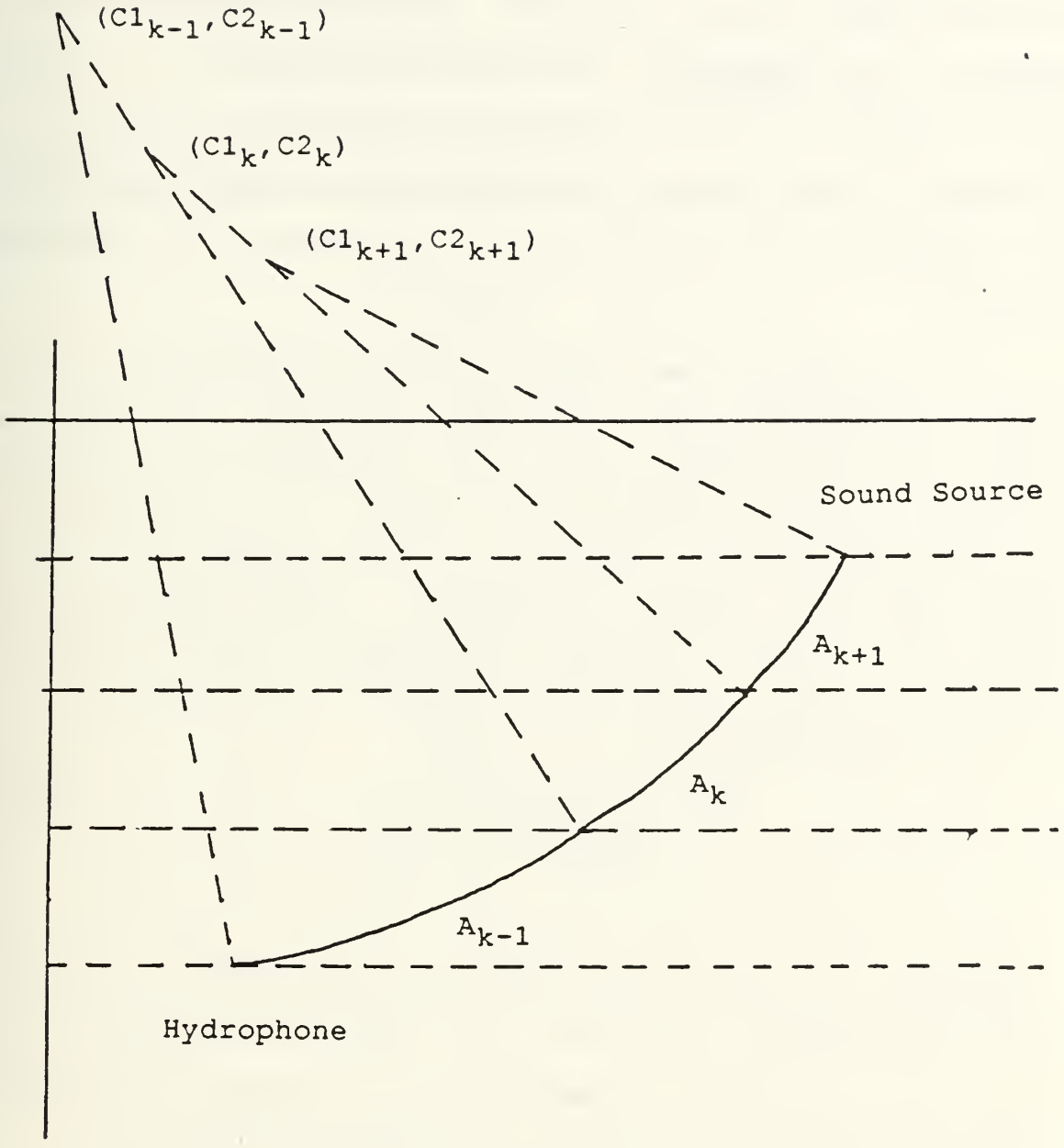


Figure 3.1 Multilayer Geometry

The algorithm is an iterative process, and needs a starting point. For the first estimate of the initial angle at the hydrophone and the time of transit, a single layer approach, using the equations presented in Chapter 2, is used. The initial velocity and gradient (V_0 and V_1) are obtained from a least squares regression of depth and velocity.

The ray path in each layer k must be calculated. To do this, the coordinates of the center of the circle of the ray path in the vertical plane containing the hydrophone and the sound source, $C1_k$ and $C2_k$, are determined. $C2_k$, which is the vertical coordinate of the center of the circle for layer k, can be calculated by adapting Equation 2.2 to a multilayer case, as follows:

$$VZ_k = V_k - (L_k)\ (G_k) \qquad (3.3)$$

$$C2_k = -VZ_k \ / \ G_k$$

Define R0 as the actual range between the hydrophone and the sound source. Let R be the range generated by the algorithm, as defined in Equation 3.1. The key calculation is to determine the location of the range coordinate of the center of the circle for each layer ($C1_k$). Once that is determined, the component of the range in each layer (R_k) can be calculated. R is initially defined to be zero. Starting at the hydrophone and iterating upward through

each layer, stopping at the depth of the sound source, $C1_k$ is calculated as follows:

$$C1_k = R - (\tan(A_k))\ (L_k - C2_{k-1}) \qquad (3.4)$$

Using Equation 3.4, the range is updated in the following manner:

$$R = C1_k + (\tan(A_{k-1}))\ (L_{k-1} - C2_{k-1}) \qquad (3.5)$$

Upon reaching the depth of the sound source, this procedure is terminated. The key observations that enable this procedure to operate is that the elevation angle at the top of one layer is identical with the elevation angle at the bottom of the layer above it, and that the radius line connecting the center of the circle used for drawing the arc in layer k ($C1_k$,$C2_k$) and the point where the arc touches the upper boundary of layer k is coincident with the radius line connecting the center of the circle for layer k-1 ($C1_{k-1}$, $C2_{k-1}$), and the point where the arc in layer k-1 intersects the lower boundary of that layer (See Figure 3.1).

The calculated range, R is then compared with the actual range. Once it is within tolerance, the procedure is terminated. If it is not, the value for the initial elevation angle A at the hydrophone must be adjusted. The ratio of the calculated range to the actual range is used to adjust the tangent of the elevation angle, and the

resulting quantity used to solve for the new value of the elevation angle at the hydrophone, as follows:

$$A_{new} = \arctan \left[\frac{(R) \tan(A_{old})}{R_{actual}} \right] \qquad (3.6)$$

Appendix B contains the FORTRAN 77 source code to implement this procedure. It requires the use of double precision throughout. The program is easily transportable to any system offering a FORTRAN 77 compiler which supports double precision operations. It has been used on both an IBM 3033 mainframe, and on a PC. Running under VM on an IBM 3033 system, the average time is approx .025 seconds to calculate a time and angle for a given set of positions.

C. ACCURACY OF ALGORITHM

The crossover zone generally occurs about 3000 feet from a sensor. In order to test the algorithm, the following procedure was adopted: a series of 200 positions in a plane containing a sensor at a depth of 1200 feet, were generated. This track started at a range of 2000 feet and at a depth of 300 feet, and extended to a range of 4000 feet and a depth of 400 feet. The positions of each point were then fed into the generation subroutine. The times and angles generated were then fed into an isogradient raytrace subroutine, and the euclidian distance between the resultant position and the actual position measured. Figure 3.2 shows the density function, the cumulative

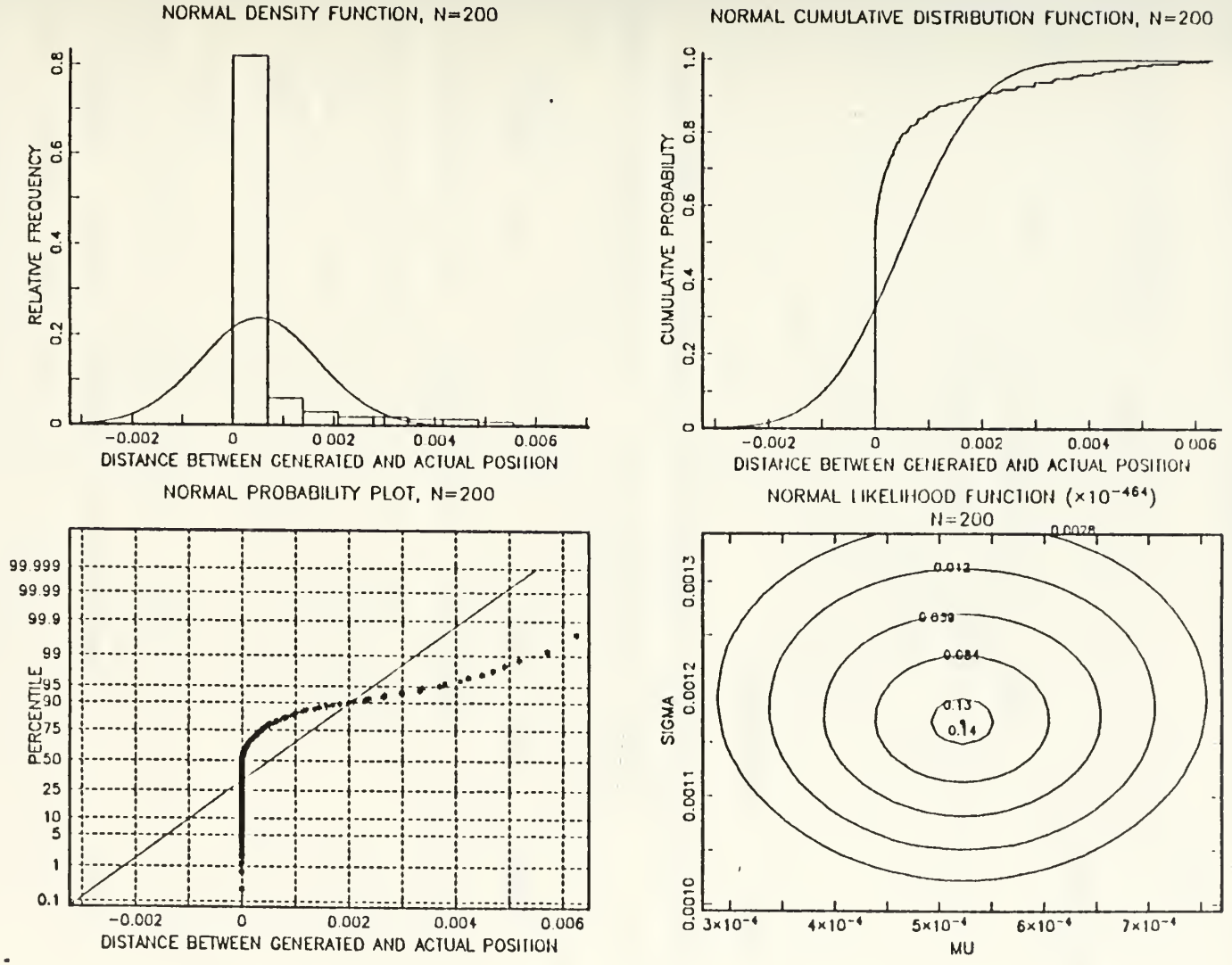


Figure 3.2 Simulation Accuracy

density function, the quantile plot and the likelihood function of the result compared to a fitted normal distribution.

The following are the relevant statistics for this distribution:

Mean = 0.0005215711385

Std Dev = 0.001172555105

Median = 0.0

IQR = 0.0003456120549

D. NANOOSE SIMULATION

The goal of this thesis was to provide NUWES with a realistic, usable simulation of the operation of their ranges to allow the exploration of "what if" cases. The actual simulation consists of a series of FORTRAN 77 programs, and are listed in appendix C. The programs were designed to be small, and manipulate data files of times and positions. This modular approach allowed easy adaption of the code to various models, which are described in chapter V.

NUWES provided a sample sound velocity profile, based on an actual set of measurements on the range. Appendix D gives the data used for this sound velocity profile. The positions of the sensors in the range coordinate system (downrange, crossrange, and depth) are given in appendix E.

Because the sound velocity profile provided extended to only 1325 feet, sensors 54 and 55 were chosen as a test

case. This was the deepest pair of adjacent sensors which fell within the limits of this profile.

E. MODEL SPECIFICS

A series of five tracks was created. Each consisted of 200 points with a horizontal spacing of ten feet. A diagram of these tracks with array number 54 designated by the triangles to the left of the track and array number 55 by the triangle to the right of the track,is shown in Figure 3.3. Following NUWES convention, each point on a track is referred to as a "point count".

Expressed in range coordinates, the starting and ending points of each track are as follows:

Track	Downrange	Crossrange	Depth
1 start	40834.00	5800.00	300.00
end	42810.00	6033.20	399.50
2 start	41710.00	6861.00	300.00
end	41941.64	4884.21	409.41
3 start	40714.00	6800.00	300.00
end	42690.00	7031.64	409.41
4 start	41000.00	6870.00	300.00
end	42562.46	5637.13	409.41
5 start	42000.00	6850.00	300.00
end	40766.35	5288.32	409.41

The actual sequence of operation for the model is as follows:

1. The positions in range coordinates for each point count are determined.

2. The range positions are converted into local coordinate positions, correcting for translation and rotation.

3. The time of travel to each hydrophone is calculated for each point count, yielding a file of 800 times per track per sensor.

4. The set of 4 times per sensor per point count is converted into an assumed position in local coordinates, and a calculated time and angle at the acoustic center of the array by using the NAVY method.

5. The positions are refined using isogradient raytracing, and then converted back into range coordinates.

In testing the range under perfect conditions, ie the "Navy Base Case", no error terms were introduced. The models described in the next section required the addition of various error terms at specified steps in this process.

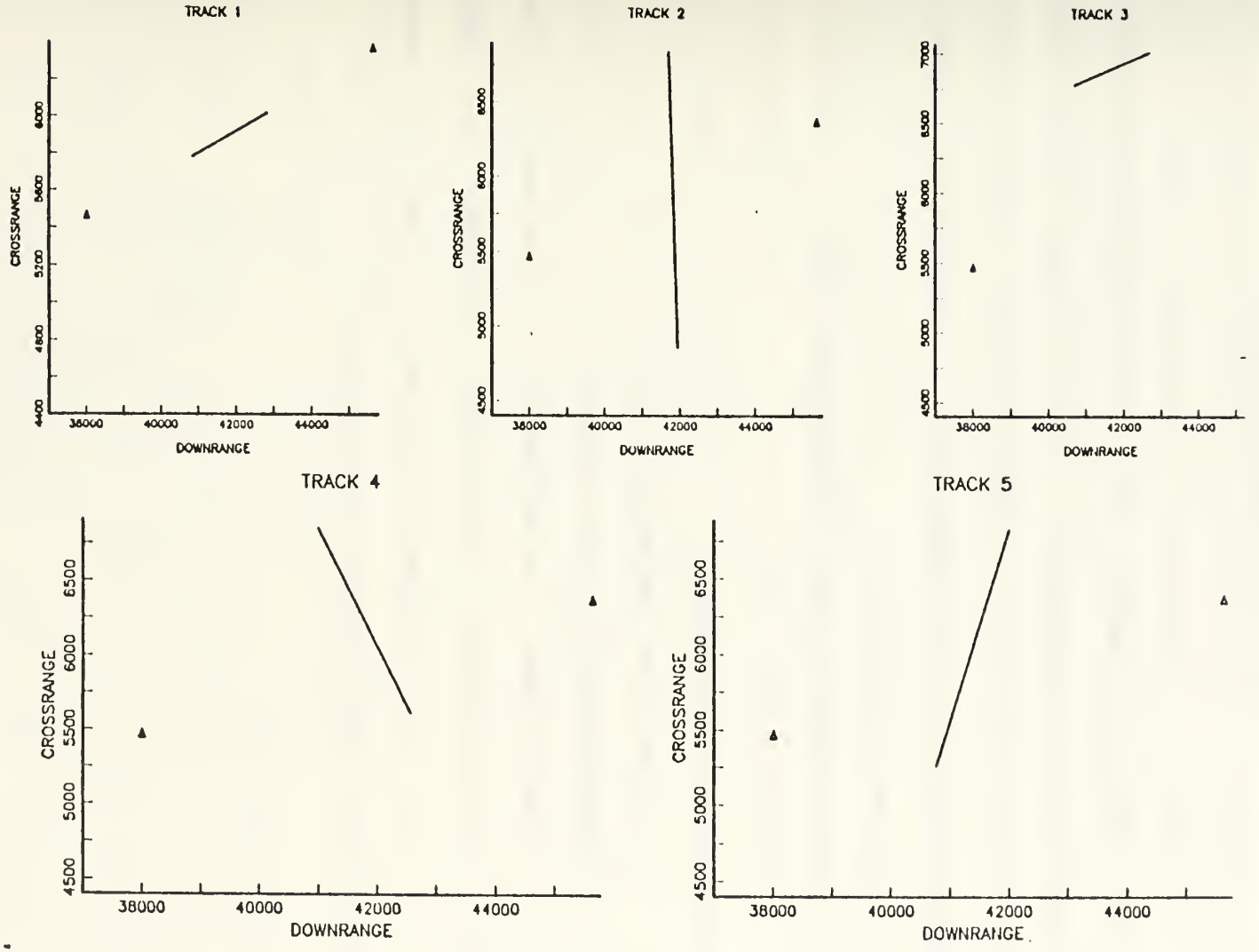


Figure 3.3 Simulation Tracks

IV. RANGE ERROR MODELS

A. OVERVIEW

The previous chapters have developed the theoretical background of the simulation, and discussed the modular implementation developed. The operation of the range was examined under five different sets of conditions, or error mechanisms, as follows:

1. No deliberate error (Navy Base Case)
2. Constant time error
3. Random time errors
4. Constant sound velocity errors.
5. Random sound velocity errors.

Each of these cases is presented in the sections that follow. The values for error terms used are for model exercise and illustration. In order to keep this thesis unclassified, actual values were not used.

B. NAVY BASE CASE

Even if no deliberate errors are introduced, there are certain assumptions in the NAVY method which lead to positional inaccuracies. In order to determine the magnitude of these errors, a no deliberate error case was investigated. Angles and times were generated using the set of five tracks discussed in the previous section, and then these generated angles and times were used as inputs

to a raytrace program, yielding a series of positions. The horizontal distance and slant range were then calculated. The results are listed in Table IV.1. Graphs of the distributions of the appear in Appendix F.

The graphs in Appendix F, as well as the graphs for the other models which appear in subsequent appendices, all graph the range (3 dimensional distance) and the distance (2 dimensional distance, excluding depth) from the final position of each point count to the actual position. This information is also presented in tabular form by Table IV.1 through Table IV.8.

Of particular interest in this table is the greater accuracy of sensor 55 as compared to sensor 54. One possible reason for this is the relative geometry of the track compared with the two sensors. Sensor 55 is looking "over its shoulder" relative to sensor 54. The exact reason for this is as yet undetermined.

Also, note that the error is significantly greater when the depth component is included. Although all four hydrophones contribute to the position information with regard to downrange and crossrange, three of the four hydrophones are at the same depth, and thus contribute nothing to depth determination. This is not a new discovery. In order to compensate for the less accurate depth resolution, the depths used on the range are based on recorded depths by a sensor on the torpedo, not by the

depths determined by the range. When comparing the horizontal distance errors, the range has a mean error of less than .15 feet.

C. CONSTANT TIME ERRORS

The entire range system is very dependent on accurate time measurement, and the maintenance of an accurate time standard throughout the range. There is a time standard in the torpedo that regulates when each ping is transmitted. Any error in synchronization of that standard adversely affects range accuracy.

The first class of errors modeled represent a constant time error. In this model, let the subscript j refer to particular point count, and the subscript k refer to the specific hydrophone (x,y,z, or c) of the sensor (either 54 or 55). Let T_{jk} be the actual time for point count j, hydrophone k. Then define T^*_{jk} as follows:

$$\text{(Case II)} \quad T^*_{jk} = T_{jk} + 0.0005 \qquad (4.1)$$

$$\text{(Case III)} \quad T^*_{jk} = T_{jk} - 0.0005$$

The results of this model are listed in Table IV.2 for Case II, and Table IV.3 for Case III. Sample tracks for Case II are illustrated in Figure 4.1, and for Case III in Figure 4.2. Graphs of the distributions are found in Appendix G and Appendix H.

D. RANDOM TIME ERRORS

The second type of timing error which can occur is an error with a random component vice the fixed bias error of cases II and III. In both cases, an error term E is added to the exact time. The error term is distributed with a normal distribution, mean = 0.0, and standard deviation = 0.00001.

In Case IV, the error term is independent from one point count to another, and is independent at each of the hydrophones for each point count. This simulates a random noise component within in system, which affects each hydrophone independently. The equation for this case is as follows:

$$\text{(Case IV)} \quad T^{*}_{jk} = T_{jk} + E_{jk} \tag{4.2}$$

Case V is similar, but the error term added to the times is the same for each hydrophone on any given point count, and independent from one point count to another. This model simulates the drift of the time synchronization of the torpedo from the range standard, vice a constant bias. The equation for time in this case is as follows:

$$\text{(Case V)} \quad T^{*}_{jk} = T_{jk} + E_{j} \tag{4.3}$$

The results for Case IV are presented in Table IV.4, and for Case V in Table IV.5. A set of sample tracks for Case IV are shown in Figure 4.3. Graphs of the distributions are found in Appendix I and Appendix J.

Tables IV.4 and IV.5 show the sensitivity of the system to errors of measurement between hydrophones. The same magnitude error term yields an average error twenty times larger if the error is independent between each hydrophone and point count than if the error is independent between each point count, but identical for each hydrophone on that point count.

E. CONSTANT VELOCITY ERROR

The second general class of error modeled is an error in the velocity of sound used for the raytracing. The first error model simulates a constant bias error in measuring the speed of sound, as might be expected from a bias error in measurement. If we let V_k be the actual velocity of sound in layer k, then the velocity used in raytracing is given by:

$$V^*_k = V_k + C \qquad (4.4)$$

In Case VI, C equals 0.005, and in Case VII, C equals 0.05. Negative values for C produced the same errors. Table IV.6 contains the data for Case VI, and Table IV.7 contains the data for Case VII. Graphs of the distributions are found in Appendix K and Appendix L. Of interest here is the relative robustness of the range to errors in velocity measurement. A tenfold increase in the error has negligible effect on the error. Comparing the values to table IV.1 shows that the overall effect of velocity errors in this range is slight.

F. RANDOM VELOCITY ERRORS

When the range is in operation, the speed of sound at depth is measured by instruments. One sounding is taken at the start of the day, at one location on the range. This is assumed to be representative of the water conditions throughout the range for the entire day.

Case VIII models what might happen if the water on the range is not homogeneous throughout the range. Let the subscript j refer to the point count, and the subscript k refer to the layer. If the actual velocity used to generate the angles and times is V_{jk}, and E_j is an error term with mean equal to 0.0 and standard deviation equal to 0.0001, then V^*_{jk} is represented by:

$$V^*_{jk} = V_{jk} + E_j \qquad (4.5)$$

Table IV.8 shows the results of this model. The graphs are contained in Appendix M. Of particular interest in this case is that the horizontal distance error appears to be highly geometry dependent. The error observed on sensor 54 is generally 7 to 9 times the error on sensor 55, when looking at the plan view (horizontal) distance. The exact reason for this is still unclear. When looking at the slant range difference, the effect of geometry is greatly reduced.

TABLE IV.1
COMPARATIVE CALCULATION ERRORS
NAVY METHOD BASE CASE

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.568	0.039	0.570	0.068	0.572
	55	0.276	0.077	0.282	0.141	0.271
2	54	0.565	0.060	0.565	0.105	0.242
	55	0.281	0.056	0.277	0.091	0.193
3	54	0.647	0.037	0.651	0.063	0.127
	55	0.195	0.076	0.194	0.126	0.193
4	54	0.596	0.013	0.595	0.025	0.065
	55	0.248	0.017	0.245	0.029	0.069
5	54	0.550	0.081	0.549	0.129	0.314
	55	0.319	0.100	0.316	0.171	0.344

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.132	0.012	0.130	0.021	0.055
	55	0.056	0.012	0.061	0.025	0.033
2	54	0.129	0.013	0.130	0.033	0.076
	55	0.057	0.010	0.061	0.013	0.036
3	54	0.145	0.014	0.139	0.016	0.054
	55	0.037	0.013	0.040	0.020	0.053
4	54	0.136	0.017	0.133	0.028	0.133
	55	0.050	0.001	0.050	0.001	0.001
5	54	0.136	0.010	0.141	0.015	0.057
	55	0.058	0.013	0.060	0.021	0.052

TABLE IV.2
CONSTANT TIME ERROR
ERROR +0.0005

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	2.489	0.010	2.487	0.019	0.034
	55	2.449	0.009	2.447	0.014	0.036
2	54	2.488	0.013	2.486	0.018	0.050
	55	2.449	0.008	2.447	0.013	0.035
3	54	2.507	0.011	2.506	0.190	0.043
	55	2.440	0.007	2.438	0.011	0.027
4	54	2.494	0.003	2.495	0.004	0.014
	55	2.443	0.003	2.442	0.004	0.012
5	54	2.485	0.017	2.482	0.026	0.072
	55	2.455	0.150	2.453	0.070	0.058

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	2.485	0.013	2.487	0.020	0.040
	55	2.303	0.005	2.306	0.010	0.010
2	54	2.488	0.013	2.486	0.190	0.049
	55	2.306	0.006	2.305	0.009	0.024
3	54	2.503	0.009	2.505	0.013	0.037
	55	2.326	0.004	2.326	0.006	0.016
4	54	2.494	0.003	2.494	0.004	0.015
	55	2.310	0.006	2.310	0.007	0.028
5	54	2.480	0.021	2.481	0.033	0.088
	55	2.312	0.003	2.312	0.004	0.016

TABLE IV.3
CONSTANT TIME ERROR
ERROR -0.0005

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	2.493	0.009	2.496	0.013	0.038
	55	2.436	0.009	2.452	0.014	0.031
2	54	2.493	0.014	2.492	0.025	0.056
	55	2.436	0.007	2.436	0.008	0.029
3	54	2.514	0.010	2.514	0.015	0.042
	55	2.428	0.006	2.428	0.008	0.027
4	54	2.500	0.004	2.500	0.006	0.023
	55	2.432	0.004	2.432	0.007	0.014
5	54	2.491	0.018	2.491	0.032	0.068
	55	2.441	0.014	2.443	0.022	0.049

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	2.220	0.035	2.225	0.055	0.137
	55	2.414	0.025	2.417	0.041	0.082
2	54	2.231	0.023	2.229	0.040	0.086
	55	2.420	0.015	2.420	0.022	0.055
3	54	2.213	0.035	2.218	0.058	0.130
	55	2.400	0.023	2.402	0.040	0.085
4	54	2.221	0.033	2.223	0.059	0.116
	55	2.412	0.009	2.413	0.016	0.034
5	54	2.207	0.003	2.207	0.006	0.016
	55	2.429	0.026	2.436	0.041	0.090

TABLE IV.4
NORMAL TIME ERRORS

Error at each hydrophone is independent for each point count

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	10.004	6.004	8.987	6.314	18.075
	55	11.301	6.305	10.542	8.337	30.996
2	54	10.444	5.890	9.708	7.520	32.251
	55	11.651	6.045	10.991	9.003	31.492
3	54	10.063	5.832	9.550	7.797	33.703
	55	12.505	7.942	10.802	9.035	43.426
4	54	10.220	5.538	8.955	7.650	25.645
	55	11.712	6.421	10.785	8.996	28.988
5	54	9.090	5.127	8.196	7.474	22.852
	55	12.668	7.461	10.647	9.006	41.698

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	7.422	5.315	6.470	5.928	31.206
	55	7.115	4.770	6.069	6.715	22.765
2	54	7.192	5.151	5.681	7.046	26.183
	55	7.504	4.932	6.526	6.685	24.192
3	54	7.025	5.118	5.754	7.001	26.167
	55	8.554	6.436	6.621	7.202	33.192
4	54	7.090	4.666	6.305	6.991	23.756
	55	7.332	5.160	5.784	7.662	24.082
5	54	6.125	4.358	5.024	5.876	20.453
	55	8.203	5.489	7.316	7.005	33.413

TABLE IV.5
NORMAL TIME ERRORS

Same error on each hydrophone for each point count

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.570	0.039	0.570	0.069	0.134
	55	0.281	0.076	0.284	0.134	0.276
2	54	0.567	0.060	0.568	0.106	0.231
	55	0.285	0.055	0.283	0.094	0.211
3	54	0.649	0.037	0.652	0.061	0.129
	55	0.201	0.074	0.201	0.129	0.270
4	54	0.597	0.012	0.599	0.018	0.068
	55	0.253	0.019	0.252	0.036	0.082
5	54	0.552	0.081	0.554	0.139	0.316
	55	0.323	0.100	0.320	0.174	0.351

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.132	0.051	0.131	0.061	0.282
	55	0.061	0.038	0.061	0.061	0.161
2	54	0.127	0.049	0.124	0.068	0.265
	55	0.063	0.040	0.060	0.061	0.202
3	54	0.139	0.048	0.138	0.069	0.278
	55	0.053	0.037	0.051	0.061	0.191
4	54	0.137	0.051	0.133	0.071	0.322
	55	0.060	0.039	0.060	0.051	0.182
5	54	0.134	0.048	0.132	0.064	0.235
	55	0.060	0.040	0.051	0.051	0.182

TABLE IV.6
CONSTANT VELOCITY ERRORS
ERROR +0.005

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.568	0.039	0.572	0.069	0.128
	55	0.277	0.077	0.280	0.141	0.273
2	54	0.565	0.060	0.565	0.103	0.234
	55	0.280	0.056	0.277	0.092	0.193
3	54	0.648	0.037	0.647	0.062	0.127
	55	0.194	0.077	0.192	0.127	0.255
4	54	0.596	0.013	0.597	0.028	0.065
	55	0.248	0.018	0.245	0.031	0.071
5	54	0.550	0.081	0.549	0.142	0.314
	55	0.319	0.100	0.316	0.171	0.342

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.135	0.011	0.132	0.011	0.045
	55	0.053	0.012	0.056	0.021	0.043
2	54	0.133	0.018	0.130	0.023	0.034
	55	0.053	0.010	0.051	0.021	0.033
3	54	0.149	0.013	0.149	0.022	0.054
	55	0.032	0.013	0.030	0.020	0.053
4	54	0.140	0.017	0.143	0.038	0.068
	55	0.049	0.003	0.050	9.0E-13	0.010
5	54	0.141	0.012	0.141	0.025	0.057
	55	0.054	0.013	0.060	0.021	0.041

TABLE IV.7
CONSTANT VELOCITY ERRORS
ERROR + 0.05

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.569	0.039	0.571	0.067	0.137
	55	0.280	0.077	0.286	0.135	0.271
2	54	0.567	0.060	0.566	0.102	0.241
	55	0.284	0.055	0.281	0.091	0.191
3	54	0.649	0.037	0.656	0.062	0.127
	55	0.200	0.074	0.200	0.130	0.258
4	54	0.598	0.012	0.596	0.018	0.065
	55	0.252	0.018	0.250	0.039	0.070
5	54	0.552	0.081	0.548	0.141	0.319
	55	0.323	0.100	0.321	0.170	0.341

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.171	0.007	0.171	9.0E-13	0.032
	55	0.017	0.006	0.020	0.010	0.020
2	54	0.168	0.017	0.171	0.034	0.074
	55	0.013	0.011	0.020	0.012	0.032
3	54	0.187	0.008	0.184	0.017	0.031
	55	0.006	0.006	0.010	0.010	0.022
4	54	0.176	0.013	0.175	0.028	0.058
	55	0.011	0.004	0.010	9.0E-13	0.022
5	54	0.173	0.015	0.173	0.027	0.080
	55	0.017	0.010	0.020	0.010	0.030

TABLE IV.8
NORMAL VELOCITY ERRORS
MEAN 0.0, STANDARD DEVIATION 0.0001

SLANT RANGE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	31.916	18.680	42.348	42.068	44.617
	55	33.214	20.513	39.959	39.036	55.999
2	54	35.360	21.347	43.324	43.364	57.573
	55	33.129	20.587	39.865	39.018	57.299
3	54	36.313	21.920	44.669	44.348	59.004
	55	32.354	20.126	38.699	38.108	56.330
4	54	31.953	18.671	42.070	42.202	45.260
	55	32.914	20.466	26.226	27.159	30.855
5	54	35.976	21.784	43.991	43.607	59.138
	55	32.570	20.167	39.412	38.957	55.549

HORIZONTAL DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	20.049	11.828	26.174	26.381	29.492
	55	2.615	1.880	3.252	3.968	5.740
2	54	20.280	11.945	26.796	26.579	29.357
	55	3.467	2.260	4.123	4.819	7.003
3	54	22.391	13.177	29.840	29.279	31.847
	55	1.840	2.104	1.080	2.073	7.021
4	54	20.378	12.073	42.070	27.159	30.854
	55	1.849	2.690	2.713	3.238	4.113
5	54	22.558	13.348	29.434	28.614	32.888
	55	3.930	2.579	4.921	5.822	7.620

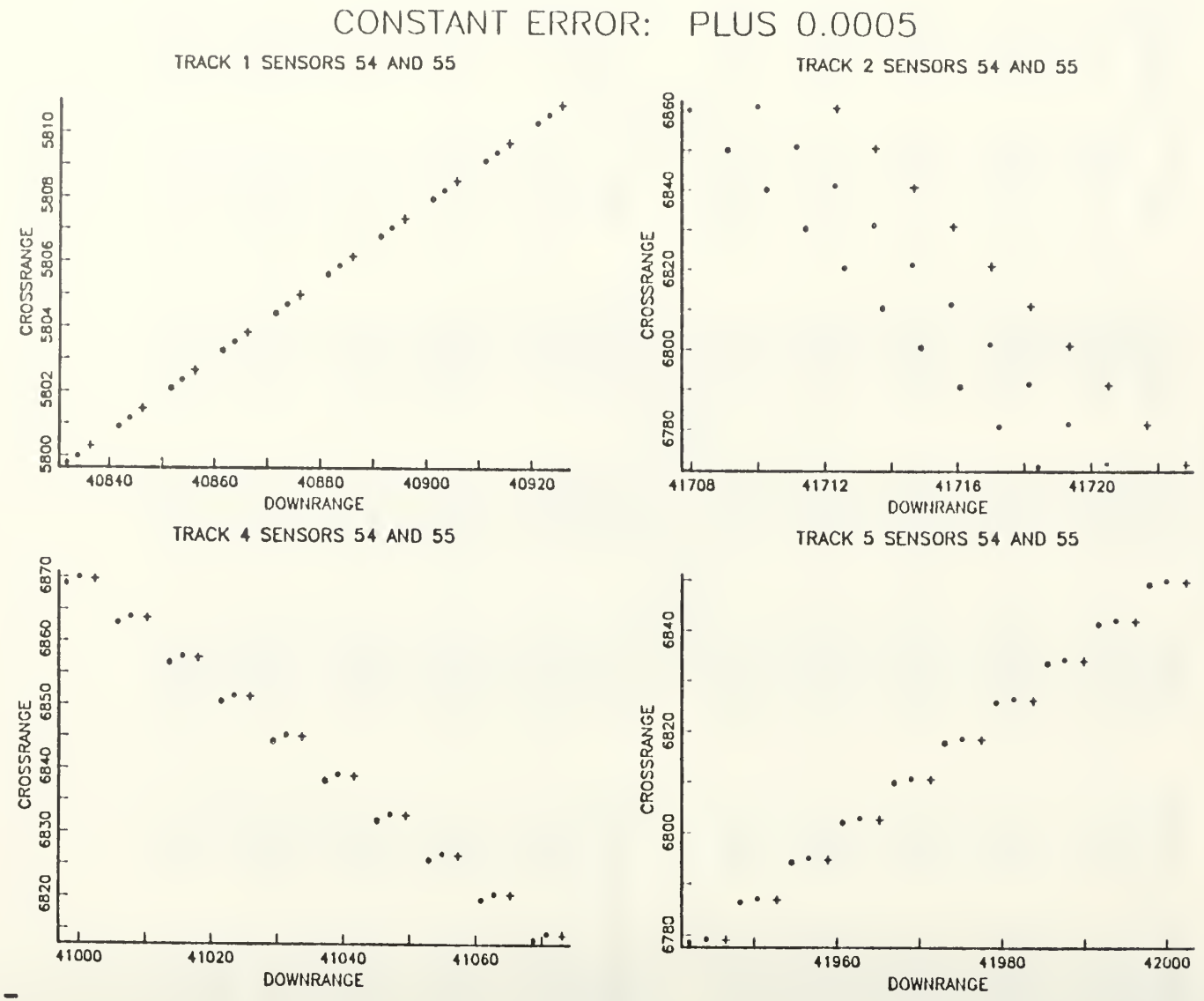


Figure 4.1 Sample Tracks: Case II

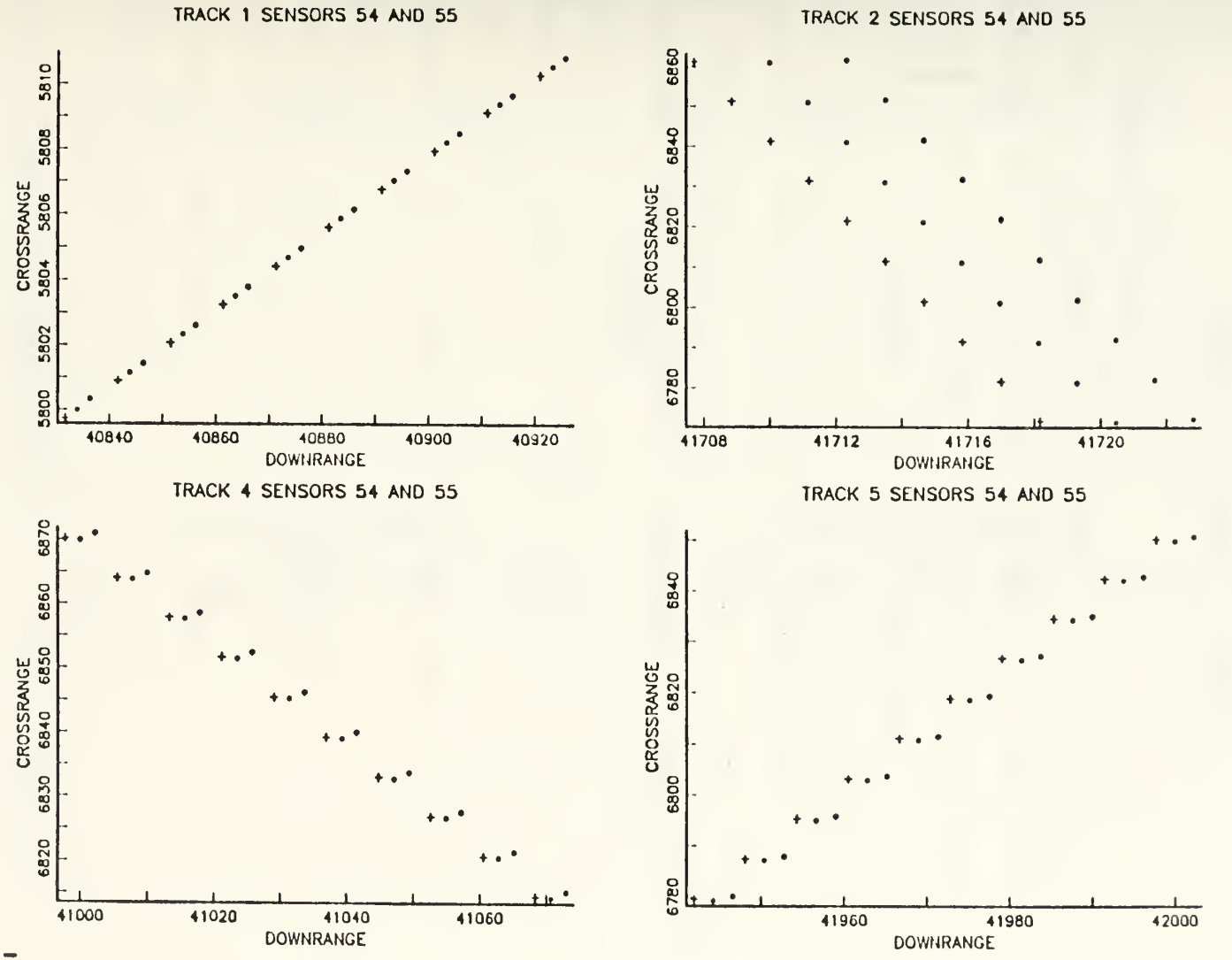


Figure 4.2 Sample Tracks: Case III

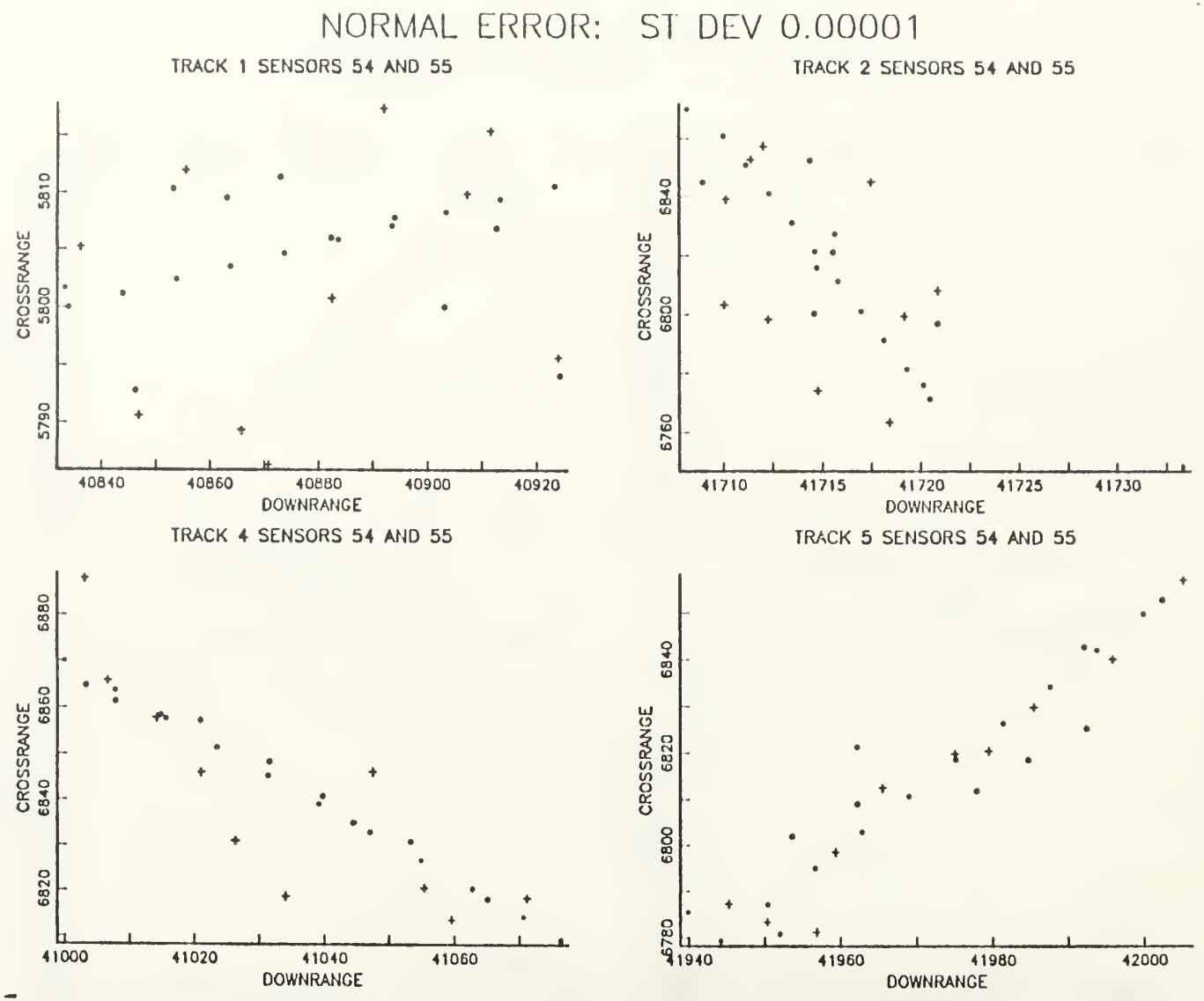


Figure 4.3 'Sample Tracks: Case IV

V. POSITION CORRECTION MODELS

A. INTERSECTION METHOD

Each of the seven error models in the preceding chapter yielded a set of tracks where neither the track from sensor 54 nor the track from sensor 55 reflected the actual position of the torpedo. The final section of this thesis is to suggest several methods for resolving this position ambiguity, and determine the "best" method was.

The intersection method yields a two dimensional (downrange and crossrange) position. Because the range uses depth as recorded by the torpedo, this is not as severe a limitation as it first appears. Two points in a plane determine a line. If, for each point count, you extend the line joining the sensor location with the final position for that particular sensor and point count until the lines from the two sensors intersect, you get a single position. The basic idea is very similar to obtaining a fix via crossed bearing lines. Table V.1 summarizes the results of this method.

As might be expected, the geometry of the track with respect to the sensor has a dramatic impact on the accuracy of this method. Track 1, which is the case where the track is head on from one sensor to another, is unreliable in all the cases. Track 3 consistently shows the best results.

This track is the one which is essentially the head on track (track 1) displaced to the North.

B. MIDPOINT METHOD

The intersection method was too geometry dependent to be of practical use, and it had the added disadvantage that the depth information was not being used. This led to the second model used, the midpoint method.

The midpoint method proceeds as follows: let (X_1, Y_1, Z_1) be the coordinates of the position for a particular point count as determined by sensor number 1. Let (X_2, Y_2, Z_2) be the coordinates for the same point count as determined by sensor number 2. The position estimate for the actual position is taken to be the midpoint of the line segment joining the two positions. The results of this method are summarized in Table V.2.

This method does not appear to be geometrically dependent. In general, this method is not as accurate as the other methods. However, it should be pointed out that it is far superior to the track extension method in Case IV, which is the normal time error where error terms are independent both between one point count and another and each hydrophone per point count.

C. DELTA METHOD

The Delta method is an attempt to combine the idea of the extension method with the concept of a fixed error.

Let P_1 and P_2 be the positions generated by sensor 1 and sensor 2 for a given point count. If you extend a fixed distance D along the line connecting the P_1 and P_2 with the location of the respective sensors C_1 and C_2, two new points P^*_1 and P^*_2 are determined.

When looking at actual range data, it appeared that many of the positional ambiguities could be resolved by this method. It was thought that there was some fixed error, either in distance or in time, that would account for the difference between positions tracked by each of the sensors. In this model, D is chosen so as to minimize the sum of the square of the distances between the new points P^*_1 and P^*_2, with the provision that D must be the same for all point counts, and both sensors.

Let P_{jk} represent the position of point count j as determined by sensor k. Let P^*_{jk} be the position which is found by taking the line joining the sensor and the respective position and extending some amount D. Define a new cartesian coordinate system with sensor k as its origin. Let T_{jk} be the angle that the line joining the origin and the position make with the positive X axis in the X Y plane, and let H_{jk} be the angle that this line makes with the positive X axis in the Y Z plane. The following equations hold from elementary trigonometry:

$$X^*_{jk} = X_{jk} + D\ \mathrm{Cos}(T_{jk}) \qquad (5.1)$$
$$Y^*_{jk} = Y_{jk} + D\ \mathrm{Sin}(T_{jk})$$
$$Z^*_{jk} = Z_{jk} + D\ \mathrm{Sin}(H_{jk})$$

The goal is to minimize the following equation:

$$\Sigma_j (X_{j1}-X_{j2})^2 + \Sigma_j (Y_{j1}-Y_{j2})^2 + \Sigma_j (Z_{j1}-Z_{j2})^2 \qquad (5.2)$$

In order to simplify the notation, define the following terms:

$$XX_j = (X_{j1} - X_{j2})$$
$$YY_j = (Y_{j1} - Y_{j2})$$
$$ZZ_j = (Z_{j1} - Z_{j2})$$
$$ST_j = \sin(T_{j1}) - \sin(T_{j2})$$
$$CT_j = \cos(T_{j1}) - \cos(T_{j2})$$
$$SH_j = \sin(H_{j1}) - \sin(H_j2)$$

A closed form solution for the minimum value of D exists. This is obtained by taking the derivative and setting it equal to zero. Using the values defined above, the solution for D is as follows:

$$D = \frac{-\ \Sigma_j\ (XX_j CT_j + YY_j ST_j + ZZ_j SH_j)}{\Sigma_j\ (ST^2_j + CT^2_j + SH^2_j)}$$

The results of the delta method are presented in tables V.3 to V.10. The first part of each table lists the data for the difference between the two positions P^*_{54} and P^*_{55}.

The second part of the table lists the data for the difference between the two generated positions and the actual position.

In general, this method provides better positions than the other methods tested. However, it does not yield one position, but two positions, one for each sensor.

TABLE V.1
INTERSECTION METHOD

Navy Base Case (Case I)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	5.321	6.911	2.363	11.059	66.474
2	0.245	2.947	0.015	0.015	41.681
3	0.011	0.005	0.010	0.006	0.021
4	0.120	0.585	0.017	0.024	7.262
5	0.085	0.609	0.017	0.034	8.518

Constant time error (Case II)
error +0.0005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	3.677	5.880	3.598	2.783	68.395
2	0.097	0.847	0.015	0.023	11.909
3	0.010	0.006	0.009	0.009	0.029
4	0.073	0.295	0.019	0.027	2.976
5	0.075	0.462	0.019	0.029	6.446

Constant Time Error (Case III)
Error -0.0005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	7.279	4.969	7.188	2.260	63.209
2	0.158	1.670	0.012	0.021	23.585
3	0.007	0.005	0.006	0.006	0.028
4	0.074	0.363	0.017	0.030	4.784
5	0.157	1.470	0.017	0.028	20.705

Normal Time errors (Case IV)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	12923.991	48843.172	3201.337	8021.191	623215.858
2	91.342	191.230	36.371	54.785	1491.058
3	22.669	14.981	18.961	21.500	62.269
4	239.197	1056.853	45.408	90.433	11439.136
5	164.426	893.113	45.628	85.742	12274.012

TABLE V.1 (CONTINUED)

Normal Time Errors (Case V)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	5.769	5.373	5.069	6.310	62.301
2	0.052	0.243	0.013	0.018	3.104
3	0.007	0.005	0.006	0.006	0.029
4	0.058	0.185	0.014	0.028	1.952
5	0.100	0.668	0.015	0.031	9.262

Constant Velocity errors (Case VI)
Error +0.005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	2.547	5.970	0.959	0.156	66.516
2	0.054	0.254	0.014	0.017	2.902
3	0.012	0.006	0.012	0.010	0.022
4	0.073	0.285	0.014	0.022	3.569
5	0.052	0.199	0.017	0.030	2.630

Constant Velocity Errors (Case VII)
Error + 0.05

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	6.155	4.108	6.165	2.026	56.807
2	0.086	0.821	0.011	0.017	11.604
3	0.005	0.004	0.005	0.005	0.035
4	0.048	0.158	0.014	0.033	1.899
5	0.096	0.807	0.016	0.029	11.398

NORMAL VELOCITY ERRORS (Case VIII)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	4.922	4.778	4.215	5.514	52.523
2	0.178	2.051	0.011	0.090	29.016
3	0.007	0.004	0.006	0.007	0.027
4	0.037	0.070	0.015	0.029	0.664
5	0.206	2.159	0.020	0.025	30.507

TABLE V.2
MIDPOINT METHOD

Navy Base Case (Case I)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	0.170	0.013	0.171	0.021	0.067
2	0.166	0.031	0.165	0.053	0.130
3	0.200	0.015	0.195	0.021	0.060
4	0.180	0.027	0.176	0.044	0.104
5	0.176	0.022	0.183	0.034	0.111

Constant Time Error (Case II)
error +0.0005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	4.879	0.020	4.883	0.025	0.045
2	4.846	0.042	4.860	0.053	0.163
3	4.797	0.017	4.803	0.020	0.074
4	4.862	0.035	4.876	0.044	0.133
5	4.852	0.025	4.861	0.029	0.106

Constant Time Error (Case III)
error -0.0005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	4.538	0.041	4.549	0.061	0.165
2	4.517	0.047	4.532	0.059	0.176
3	4.400	0.042	4.414	0.066	0.155
4	4.502	0.081	4.528	0.134	0.275
5	4.500	0.036	4.517	0.051	0.132

Normal Time Error (Case IV)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	11.323	8.589	9.825	9.736	52.081
2	11.754	7.683	10.229	10.192	39.937
3	11.778	7.879	9.875	10.190	41.315
4	11.532	7.408	10.842	10.457	42.982
5	9.726	9.737	8.155	9.276	33.325

TABLE V.2 (CONTINUED)

Normal Time Errors (Case V)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	0.171	0.078	0.177	0.106	0.407
2	0.165	0.075	0.162	0.106	0.410
3	0.192	0.075	0.187	0.107	0.451
4	0.181	0.077	0.174	0.107	0.469
5	0.175	0.077	0.171	0.111	0.350

Constant Velocity Errors (Case VI)
Error +0.005

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	0.177	0.011	0.176	0.015	0.052
2	0.174	0.031	0.170	0.045	0.136
3	0.209	0.013	0.209	0.024	0.055
4	0.187	0.025	0.190	0.153	0.100
5	0.185	0.025	0.182	0.048	0.106

Constant Velocity Error (CAse VII)
Error +0.05

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	0.249	0.008	0.247	0.005	0.040
2	0.244	0.030	0.247	0.567	0.122
3	0.282	0.009	0.276	0.016	0.032
4	0.259	0.022	0.257	0.043	0.087
5	0.252	0.027	0.249	0.045	0.132

Normal Velocity Error (Case VIII)

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	29.186	17.192	39.018	37.722	41.691
2	28.756	16.987	37.777	37.962	42.789
3	33.887	20.014	44.315	44.705	48.712
4	29.660	17.550	38.259	39.151	44.968
5	31.897	18.893	41.439	41.694	46.940

TABLE V.3
NAVY BASE CASE (CASE I)

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.008	0.004	0.009	0.001	0.041	-0.036
2	0.033	0.018	0.031	0.027	0.102	-0.034
3	0.042	0.007	0.040	0.010	0.031	-0.051
4	0.028	0.015	0.023	0.025	0.077	-0.040
5	0.033	0.020	0.029	0.033	0.167	-0.028

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.097	0.012	0.095	0.021	0.057
	55	0.092	0.012	0.096	0.025	0.033
2	54	0.094	0.018	0.096	0.033	0.080
	55	0.091	0.010	0.095	0.013	0.036
3	54	0.094	0.014	0.088	0.016	0.054
	55	0.088	0.014	0.091	0.020	0.051
4	54	0.095	0.017	0.092	0.028	0.069
	55	0.091	0.001	0.091	0.001	0.001
5	54	0.109	0.010	0.114	0.015	0.060
	55	0.086	0.013	0.088	0.021	0.053

TABLE V.4
CONSTANT TIME ERROE (CASE II)
Error +0.0005

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.013	0.008	0.011	0.011	0.043	-2.392
2	0.026	0.015	0.026	0.027	0.092	-2.396
3	0.043	0.004	0.043	0.005	0.023	-2.412
4	0.021	0.013	0.018	0.022	0.056	-2.400
5	0.033	0.020	0.027	0.032	0.111	-2.385

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.094	0.013	0.096	0.020	0.040
	55	0.088	0.004	0.085	0.009	0.009
2	54	0.092	0.013	0.091	0.019	0.062
	55	0.089	0.006	0.090	0.009	0.025
3	54	0.090	0.009	0.093	0.013	0.038
	55	0.086	0.004	0.086	0.006	0.016
4	54	0.094	0.003	0.094	0.004	0.019
	55	0.090	0.005	0.090	0.007	0.028
5	54	0.096	0.022	0.097	0.033	0.103
	55	0.073	0.003	0.073	0.004	0.016

TABLE V.5
CONSTANT TIME ERROR (CASE III)
ERROR -0.0005

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.013	0.010	0.009	0.009	0.060	2.319
2	0.040	0.023	0.041	0.038	0.104	2.327
3	0.051	0.107	0.049	0.025	0.069	2.309
4	0.035	0.022	0.028	0.030	0.095	2.319
5	0.036	0.022	0.032	0.035	0.116	2.329

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.094	0.035	0.945	0.546	0.144
	55	0.095	0.025	0.097	0.041	0.082
2	54	0.096	0.022	0.097	0.040	0.101
	55	0.093	0.015	0.094	0.022	0.055
3	54	0.096	0.035	0.091	0.056	0.132
	55	0.091	0.023	0.094	0.040	0.086
4	54	0.098	0.033	0.096	0.059	0.121
	55	0.093	0.009	0.094	0.160	0.033
5	54	0.123	0.004	0.122	0.006	0.036
	55	0.100	0.026	0.107	0.041	0.090

TABLE V.6
NORMAL TIME ERROR (CASE IV)

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	9.813	7.414	7.961	9.583	41.164	0.252
2	10.765	6.650	9.873	9.875	28.696	0.113
3	11.882	7.187	10.843	9.536	35.522	0.210
4	10.571	6.947	8.840	9.343	37.944	-0.039
5	9.797	6.475	8.185	8.750	32.991	0.409

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	7.425	5.314	6.467	5.851	31.063
	55	7.104	4.770	6.027	6.726	22.962
2	54	7.194	5.152	5.685	7.071	26.303
	55	7.501	4.394	6.529	6.774	24.231
3	54	7.041	5.105	5.780	7.010	26.178
	55	8.557	6.430	6.637	7.067	33.185
4	54	7.089	4.666	6.307	6.998	23.754
	55	7.333	5.160	5.779	7.640	24.039
5	54	6.117	4.360	4.947	5.774	20.601
	55	8.200	5.485	7.408	7.029	33.254

TABLE V.7
NORMAL TIME ERRORS (CASE V)

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.057	0.040	0.048	0.059	0.207	-0.036
2	0.062	0.039	0.055	0.047	0.233	-0.033
3	0.076	0.036	0.071	0.039	0.274	-0.047
4	0.062	0.041	0.056	0.048	0.220	-0.040
5	0.064	0.044	0.054	0.058	0.199	-0.029

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.097	0.049	0.096	0.061	0.250
	55	0.092	0.044	0.097	0.071	0.191
2	54	0.095	0.047	0.090	0.068	0.228
	55	0.092	0.045	0.094	0.094	0.061
3	54	0.023	0.047	0.091	0.069	0.238
	55	0.089	0.048	0.088	0.071	0.234
4	54	0.097	0.050	0.093	0.071	0.300
	55	0.094	0.046	0.100	0.061	0.218
5	54	0.106	0.048	0.103	0.064	0.225
	55	0.085	0.043	0.078	0.051	0.202

TABLE V.8
CONSTANT VELOCITY ERRORS (CASE VI)
ERROR +0.005

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.006	0.005	0.003	0.007	0.038	-0.039
2	0.033	0.017	0.032	0.029	0.101	-0.039
3	0.042	0.007	0.041	0.009	0.031	-0.057
4	0.027	0.015	0.020	0.021	0.080	-0.044
5	0.034	0.021	0.028	0.040	0.103	-0.032

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.097	0.011	0.093	0.011	0.048
	55	0.092	0.012	0.095	0.021	0.023
2	54	0.094	0.018	0.092	0.023	0.087
	55	0.091	0.010	0.089	0.021	0.033
3	54	0.094	0.013	0.093	0.022	0.054
	55	0.088	0.013	0.085	0.020	0.051
4	54	0.097	0.017	0.010	0.038	0.069
	55	0.092	0.003	0.094	0.0001	0.011
5	54	0.109	0.012	0.110	0.025	0.060
	55	0.086	0.013	0.092	0.021	0.041

TABLE V.9
CONSTANT VELOCITY ERRORS (CASE VII)

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	0.006	0.005	0.003	0.009	0.032	-0.075
2	0.033	0.020	0.033	0.034	0.103	-0.074
3	0.048	0.005	0.048	0.006	0.028	-0.093
4	0.026	0.017	0.023	0.029	0.073	-0.081
5	0.033	0.022	0.022	0.034	0.116	-0.067

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	0.096	0.007	0.096	1.0E-9	0.036
	55	0.092	0.006	0.095	0.010	0.020
2	54	0.094	0.017	0.097	0.034	0.078
	55	0.091	0.011	0.094	0.012	0.032
3	54	0.094	0.008	0.091	0.017	0.033
	55	0.089	0.008	0.093	0.010	0.030
4	54	0.096	0.013	0.094	0.028	0.060
	55	0.092	0.004	0.091	3.0E-5	0.020
5	54	0.106	0.015	0.106	0.026	0.086
	55	0.083	0.010	0.087	0.010	0.030

TABLE V.10
NORMAL VELOCITY ERRORS (CASE VIII)

DELTA METHOD: TWO DIMENSIONAL DIFFERENCE

TRK	MEAN	ST DEV	MEDIAN	IQR	RANGE	DELTA
1	9.798	5.300	7.666	12.167	22.799	-8.865
2	9.843	4.570	8.971	10.091	25.051	-8.353
3	14.219	5.289	12.447	11.919	26.105	-11.419
4	10.005	5.281	7.446	11.406	27.319	-9.208
5	10.059	5.617	9.037	12.233	28.710	-9.351

DELTA METHOD: TWO DIMENSIONAL MISS DISTANCE

TRACK	SENSOR	MEAN	ST DEV	MEDIAN	IQR	RANGE
1	54	15.755	4.243	17.240	8.522	16.090
	55	10.659	2.743	10.641	4.692	10.836
2	54	16.275	4.839	18.470	9.931	16.464
	55	11.782	2.316	12.477	4.950	9.723
3	54	16.938	3.481	18.454	6.503	17.768
	55	10.673	2.696	11.479	1.550	9.184
4	54	15.966	4.419	17.061	8.821	17.405
	55	11.008	1.696	10.277	3.372	7.764
5	54	18.084	5.395	20.103	9.954	16.129
	55	13.250	2.626	14.272	5.928	9.879

VI. CONCLUSIONS AND RECOMMENDATIONS

A. GENERAL

The goal of this thesis was to provide a usable, realistic simulation of the operation of the ranges under NUWES control, and to explore some "what if" cases relating to sources of error on the range, and means of resolving positional ambiguity. The simulation operates as desired. Several error models were explored, and three models to resolve the true position of the torpedo were examined.

The values used for bias error and standard deviations of error components were chosen to be illustrative vice predictive. The use of unsanitized values, while presenting no problems to the operation of the simulation, would have required the classification of this report.

One sound velocity profile was available for my use in this thesis. There was no track identification corresponding to this profile, nor was there any time/date information. These factors led me to explore several possible models in using generic values, rather than do extensive analysis on a single model. Despite this, some general conclusions are possible.

B. RANGE OPERATION AND ERROR MECHANISMS

The geometry of the sensor and the torpedo appears to have an effect on the accuracy of the track. Although this

effect is small, and easily masked by other sources of error, it exists. Specifically, in the base case, sensor 55 had an average error only one third the size of sensor 54. Sensor 55 (depth 1215.0) is slightly deeper than sensor 54 (depth 1188.1). Tests of the algorithm run at differing depths showed no appreciable difference in performance. This difference may be due to fact that the assumptions of the NAVY method tend to increase the error as the curvature of the ray path increases.

The system seems to very predictable with respect to constant bias errors in time. Random errors have a much greater effect, especially when an error affects one hydrophone more than another for a given point count. Any timekeeping error which would tend to effect one of the hydrophones more than another would have great effect on accuracy. Elimination of these errors will have a far greater effect than elimination of constant errors of the same magnitude, or random errors which effect all hydrophones equally on a given point count.

Constant velocity errors, at least in the values studied in this report, have very little effect on the accuracy of the range. Random velocity errors have a large effect, but again this was different between sensor 54 and 55. The reasons for this are as yet unclear.

C. INTERSECTION METHOD

The intersection method is highly geometry dependent. For some as yet undetermined reason it was superior to all other methods in the random velocity error case. In the case of timing errors, it is less reliable. As the displacement from the actual track increases in a random fashion, such as encountered in the random time error models, this method becomes less and less useful. If the geometry of the situation is favorable, it can give a good approximation, but other methods offer much better, more consistent accuracy for the same computational effort.

D. MIDPOINT METHOD

This method is the easiest to employ. It is less geometry dependent than the intersection method, but it is far less reliable. It is the least promising of all the methods tested, and does not warrant further study.

E. DELTA METHOD

The delta method is the most promising method of all three tested. It was not geometry dependent, and consistently resulted in more accurate position estimates than the other methods.

The big disadvantage of the delta method is that it still yields two positions, not one. However, each of these position estimates are considerably more accurate than the original position.

F. RECOMMENDATIONS FOR FURTHER RESEARCH

Several areas of interest need further exploration. First of all, we need to develop a better understanding of the errors inherent in the Navy position method. The operation of the range under varying sound velocity profiles needs to be explored. Once this is completed, a better understanding of the shape and characteristics of the error function of the range can be developed.

The simulation will be of benefit in testing outlier rejection techniques. NUWES is now studying methods of outlier rejection. This simulation will provide an excellent means to study the effect of these methods against known positions.

Further refinements to the delta position method are needed. One area which initially appears promising is to consider the constant to be added to each raytrace as a constant time, not a constant distance. This method should consider the time along the ray path, not the straight line used presently. Based on current results, this improved delta method should be extremely accurate for constant time bias errors, and be better than the existing method for other error cases.

Computer Program for Isogradient Raytracing

```
C*************************************************************
C
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C      THIS SUBROUTINE EXECUTES ISOGRADIENT RAYTRACING FOR A
C     53 LEVEL PROBLEM.
C
C     CALLING ARGUMENTS ARE AS FOLLOWS:
C     TT0 -- TRANSIT TIME, MEASURED IN SECONDS
C     AA --  ANGLE AT SENSOR, MEASURED IN RADIANS
C     D --  DEPTH OF SENSOR, (POSITIVE DOWN)
C     VEL -- ARRAY CONTAINING SOUND VELOCITY AT EACH LAYER
C     L -- ARRAY CONTAINING THE DEPTH OF THE LAYER MIDPOINTS
C     G -- THE GRADIENT WITHIN EACH LAYER
C
C     VALUES RETURNED ARE AS FOLLOWS:
C     R -- THE HOROZONTAL RANGE OF THE TARGET FROM THE SENSOR
C     Z -- THE DEPTH OF THE TARGET
C
C     ALL DEPTHS POSITIVE DOWNWARD
C     FLOATING PT NUMBERS ARE REAL*8
C
C*************************************************************

      SUBROUTINE GTRC(TT0,AA,D,VEL,Z,R,L,G)
      DIMENSION VEL(53),L(53),G(53)
      REAL*8 VEL,T0,A,Z,R,L,DL,G,DZ,DR,V0,C1,C2,T,DT
      REAL*8 DSIG,A1,D,RV,RS,TT0,AA
      REAL*8 EP,X,Y,X0,GG,GP

C*************************************************************
C
C      INITIALIZE ALGORITHM.  FIND THE LAYER THAT CONTAINS THE
C     SENSING ARRAY AND THE DISTANCE TO THE TOP OF THAT
C     LAYER.
C
C     A AND T0 ARE USED WITHIN THE SUBROUTINE FOR ANGLE AND
C     TIME SO THAT THE ORIGINAL VALUES ARE NOT CHANGED IN
C     THE MAIN PROGRAM.  DL SETS THE LAYER DEPTH.  DZ IN THE
C     INCREMENT IN DEPTH.  R IS THE RANGE, AND T IS TIME.
C
C*************************************************************

      A=AA
      T0=TT0
      DL=25.0D0
      I=0
```

```
      DO 20 J=1,52
      IF((L(J).LT.D).AND.(D.LE.L(J+1))) I=J
   20 CONTINUE
      DZ=D-L(I)
      Z=D
      R=0.0D0
      T=0.0D0

C*************************************************************
C
C     COMPUTE SPEED AT LAYER BOTTOM, COMPUTE RV, THE RAY
C     INVARIANT,AND COMPUTE THE INITIAL LAYER TRANSIT TIME.
C
C*************************************************************

      RS=VEL(I) + (G(I)*DZ)
      RV=(DCOS(A))/RS

C*************************************************************
C
C     C1 IS THE HOROZONTAL COMPONENT OF THE CENTER OF
C     THE CIRCLE.  C2 IS THE VERTICAL COMPONENT OF THE CENTER
C     A1 IS THE EXIT ANGLE OF THE LAYER.  DT IS THE TIME
C     INCREMENT IN THE LAYER.  DR IS THE RANGE INCREMENT IN
C     THE LAYER
C
C*************************************************************

  100 V0=VEL(I)-(G(I)*L(I))
      C2=-V0/G(I)
      C1=(C2-Z) * DTAN(A)
      RS=DSQRT(C1*C1 + ( (Z-C2)*(Z-C2)))
      A1=DACOS(RV*VEL(I))
      DT=DLOG((DCOS(A)*(1.0D0+DSIN(A1)))/(DCOS(A1)*
     1(1.0D0+DSIN(A))))
      DT=DT/G(I)
      IF (C1.LT.0.0D0) THEN
         DSIG = -1.0D0
      ELSE
         DSIG = 1.0D0
      END IF
      IF (C1.EQ.0.0D0)  DSIG = 0.0D0
      DR=C1-(DSIG*RS*DSIN(A1))

C*************************************************************
C
C     UPDATE AND TEST
C
C*************************************************************

      Z=Z-DZ
      R=R+DR
```

```
      T=T+DT
      DZ=DL
      A=A1
      I=I-1
      IF (I.EQ.0) THEN
         WRITE(*,200)
  200     FORMAT(//,' THE POSITION IS ABOVE THE SURFACE FOR THE
INPUTS')
         RETURN
      END IF
      IF(T.LT.T0) GO TO 100

C*************************************************************
C
C     FINALIZE.  THE ALGORITHM CALCULATES RANGE AND TIME
C     LAYER BY LAYER.  THIS SECTION OF CODE CORRECTS FOR
C     THE OVERSHOOT IN THE LAST LAYER, AND THEN CALCULATES
C     THE EXIT ANGLE USING NEWTON-RAPHSON ITERATION
C
C*************************************************************

      DT=T0-T
      R=R-DR
      Y=(G(I+1)*DT)-DLOG( (DCOS(A))/(1.0D0+DSIN(A)))
      X=DEXP(Y)-1.0D0
      EP=.0000000001D0

C*************************************************************
C
C     START ITERATION
C
C*************************************************************

    5 X0=X
      GG=DEXP(Y)*DCOS(X)-DSIN(X)-1.0D0
      GP=-DCOS(X)-DEXP(Y)*DSIN(X)
      X=X0-GG/GP

C*************************************************************
C
C     TEST
C
C*************************************************************

      IF (EP.LT.DABS(GG)) GO TO 5
      A1=X
      Z=C2+RS*DCOS(A1)
      IF (C1.LT.0.0D0) THEN
         DSIG = -1.0D0
      ELSE
         DSIG = 1.0D0
      END IF
```

```
IF (C1.EQ.0.0D0)  DSIG = 0.0D0
DR=C1-(DSIG*RS*DSIN(A1))
R=R+DR
RETURN
END
```

APPENDIX B

Computer Program for Time and Angle Generation

```
C*************************************************************
C
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C     THIS SUBROUTINE GENERATES TRANSIT TIME AND ELEVATION
C     ANGLE AT A SENSOR IF GIVEN THE HOROIZONTAL RANGE,
C     THE DEPTH OF THE SENSOR AND THE TARGET, THE LAYER
C     BOUNDARIES AND THE GRADIENTS
C
C     CALLING ARGUMENTS ARE AS FOLLOWS:
C     LL -- AN ARRAY CONTAINING THE LAYER MIDPOINTS
C     G --  AN ARRAY CONTAINING THE GRADIENTS FOR EACH LAYER
C     VV -- AN ARRAY CONTAINING THE VELOCITY AT EACH LAYER
C     A2 -- THE DEPTH OF THE SENSOR (POSITIVE DOWN)
C     P1 -- RANGE OF THE TARGET (HORIZONTAL RANGE)
C     P2 -- DEPTH OF THE TARGET (POSITIVE DOWN)
C     V0, V1 --THE VALUES FOR A STRAIGHT LINE SINGLE LAYER
C       REGRESSION OF DEPTH VRS VELOCITY
C     DEPTH -- AN ARRAY CONTAINING THE DEPTH OF EACH LAYER
C
C     RETURN ARGUMENTS ARE AS FOLLOWS:
C     ANGLE -- THE FINAL ANGLE AT THE SENSOR
C     TIME -- THE FINAL TIME OF TRANSIT
C
C     ALL FLOATING POINT NUMBERS ARE REAL*8
C     ALL TIME IN SECONDS AND ALL ANGLES IN RADIANS
C
C*************************************************************

      SUBROUTINE TGEN(LL,G,VV,A2,P1,P2,ANGLE,
     1TIME,V0,V1,DEPTH)
      DIMENSION L(53),G(52),V(53),LL(53),VV(53)
      DIMENSION TH(53),T(53),VZ(53)
      DIMENSION C2(53),TT(53),DEPTH(53)
      REAL*8 L,G,LL,VV,TH,T,VZ,C2,TT,R0,A1
      REAL*8 A2,P1,P2,C1,C22,THETA
      REAL*8 THETAZ,RV,R,TIME,ANGLE,EP,DZ,DEPTH,V0,V1,V
```

```
C*************************************************************
C
C     INITIALIZE.  SET VALUE FOR DZ, THE LAYER THICKNESS.
C     SENSOR IS ASSUMED TO BE AT RANGE 0.  DETERMINE THE
C     VALUES FOR J, WHICH IS 1+NUMBER OF LAYERS  LESS THAN
C      OR EQUAL TO SENSOR DEPTH, AND I, WHICH IS THE NUMBER OF
C     LAYERS LESS THAN OR EQUAL TO TORPEDO DEPTH.  REDEFINE
C     THE ENDPOINTS OF THOSE LAYERS LOCALLY TO BE THE
C     DEPTHS OF THE TORPEDO AND SENSOR.  DEFINE LOCAL VALUES
C     FOR THE LL VV ARRAYS.
C
C*************************************************************

      EP=0.1D-6
      DZ=25.0D0
      A1=0.0D0
      J=1
      I=0
      DO 10 K=1,53
         L(K)=LL(K)
         V(K)=VV(K)
         IF(DEPTH(K).LE.A2) J=J+1
         IF(DEPTH(K).LE.P2) I=I+1
   10 CONTINUE
      N=1+J-I
      V(I)=V(I)+G(I)*(P2-L(I))
      V(J)=V(J-1)+G(J-1)*(A2-L(J-1))
      L(I)=P2
      IF(A2.GT.L(J-1)) G(J)=G(J-1)
      L(J)=A2
      R0=P1-A1

C*************************************************************
C
C     CALCULATE AN INITIAL ESTIMATE FOR THE ANGLE AND TIME
C     USING A SINGLE LAYER APPROXIMATION
C
C*************************************************************

      C22=-V0/V1
      C1=((0.5D0)*(P1-A1))
      C1=C1+((0.5D0)*(L(I)-L(J))*(L(I)+L(J)-2.0D0*
     1C22)/(P1-A1))
      THETA=DATAN((A1-C1)/(A2-C22))
```

```
C*************************************************************
C
C     USE THE ANGLE THETA TO RAYTRACE BACK THROUGH ALL THE
C     LAYERS.  FIRST, USE THE RAY INVARIANT (RV) AND THE
C     VELOCITY TO CALCULATE THE ENTRANCE ANGLE AT EACH
C     LAYER.
C
C*************************************************************
      R=A1
   50 RV=DCOS(THETA)/V(J)
      DO 60 K=I,J
         TH(K)=DACOS(RV*V(K))
         T(K)=DTAN(TH(K))
         VZ(K)=V(K)-L(K)*G(K)
         C2(K)=-VZ(K)/G(K)
   60 CONTINUE
C*************************************************************
C
C     USING THE ANGLE JUST CALCULATED, ITERATE BACKWARDS
C     THROUGH THE LAYERS FROM SENSOR TO TARGET TO GET THE
C     HORIZONTAL RANGE.  STOP AT THE DEPTH OF THE TARGET.
C
C*************************************************************
      R=0.0D0
      DO 70 K=J,I+1,-1
         C1=R-T(K)*(L(K)-C2(K-1))
         R=C1+T(K-1)*(L(K-1)-C2(K-1))
   70 CONTINUE
C*************************************************************
C
C     TEST IF THE VALUE FOR THE RANGE IS WITHIN TOLERANCE.
C     IF NOT, REDEFINE THETA, THE INITIAL ANGLE, AND
C     RAYTRACE AGAIN.  IF WITHIN TOLERANCE, CALCULATE TIME
C     OF TRAVEL BASED ON THETA, AND RETURN.
C
C*************************************************************
      IF((DABS(R-P1)).LE.EP) GOTO 100
      THETAZ=THETA
      THETA=DATAN(DTAN(THETAZ)*(R-A1)/R0)
      GOTO 50
  100 TT(J)=DLOG((1.0D0+DSIN(TH(J)))/(DCOS(TH(J))))
      TIME=0.0D0
      DO 110 K=J-1,I,-1
      TT(K)=DLOG((1.0D0+DSIN(TH(K)))/(DCOS(TH(K))))
      TIME=TIME+(TT(K)-TT(K+1))/G(K)
  110 CONTINUE
      ANGLE=THETA
      RETURN
      END
```

APPENDIX C

Simulation Programs

```
C*************************************************************
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C      THIS PROGRAM CONVERTS A USER DESIGNATED INPUT FILE IN
C      GLOBAL RANGE COORDINATES TO A FILE IN LOCAL COORDINATES
C     FOR ANY DESIRED ARRAY.
C
C     INPUT FILE 40 IS THE INPUT FILE FOR ALL SENSORS.  EACH
C     LINE OF THE INPUT FILE CONSISTS OF THE SENSOR NUMBER,
C     THE DOWNRANGE, CROSSRANGE, AND DEPTH OF THE SENSOR,
C     AND THE VALUES OF X-TILT, Y-TILT, AND Z-ROT.
C
C     INPUT FILE 44 IS THE GLOBAL POSITION OF THE TORPEDO.
C     EACH LINE OF THE INPUT FILE CONSISTS OF THE DOWNRANGE,
C     CROSSRANGE, AND DEPTH, ONE LINE PER POINT COUNT
C
C     OUTPUT FILE 41 IS THE OUTPUT FILE OF LOCAL POSITIONS
C     THE FIRST LINE WILL BE THE SENSOR NUMBER, SENSOR
C     DEPTH, AND INDEX NUMBER (WHICH CAN BE IGNORED)
C     EACH SUBSEQUENT LINE IS THE DOWNRANGE, CROSSRANGE, AND
C     DEPTH COORDINATE FOR THE TRACK IN THE LOCAL SYSTEM.
C
C     ALL CALCULATIONS ARE MADE USING REAL*4 VARIABLES.
C
C     THE SENSOR NUMBER DESIRED IS ENTERED INTERACTIVELY BY
C     THE USER.
C
C*************************************************************

      PROGRAM CONVRT
      DIMENSION LOCAL(200,3), GLOBAL(200,3), POSIT(50,6)
    DIMENSION JNR(50)
      REAL*4 LOCAL

C*************************************************************
C
C     READING IN THE INPUT FILES
C
C*************************************************************

      DO 10 J=1,50
         READ(40,205,END=5) JNR(J),(POSIT(J,K),K=1,6)
  205    FORMAT(I2,6F10.1)

   10 CONTINUE
    5 DO 20 J=1,200
         READ(44,210,END=50) (GLOBAL(J,K),K=1,3)
  210    FORMAT(10X,3(5X,F10.2))
```

```
   20 CONTINUE

C*************************************************************
C
C     INTERACTIVE ENTRY OF THE SENSOR NUMBER
C
C*************************************************************

   50 WRITE(*,115)
  115 FORMAT(/,' ENTER NUMBER OF THE SENSOR',/)
      READ(*,215) I
  215 FORMAT(I2)
      K=-1
      DO 55 J=1,50
         IF(JNR(J).EQ.I) K=J
   55 CONTINUE
      IF(K.LT.0) GOTO 50

C*************************************************************
C
C     DETERMINING THE PROPER VALUES FOR THE CONVERSION
C
C*************************************************************

      XC=POSIT(K,1)
      YC=POSIT(K,2)
      ZC=POSIT(K,3)
      XTILT=POSIT(K,4)
      YTILT=POSIT(K,5)
      ZROT=POSIT(K,6)
      C1=COS(YTILT)
      S1=SIN(YTILT)
      C2=COS(XTILT)
      S2=SIN(XTILT)
      C3=COS(ZROT)
      S3=SIN(ZROT)

      DO 60 J=1,200

C*************************************************************
C
C     CONVERTING FOR LATERAL DISPLACEMENT
C
C*************************************************************

         X=GLOBAL(J,1)-XC
         Y=GLOBAL(J,2)-YC
         Z=GLOBAL(J,3)
```

```
C*************************************************************
C
C     CONVERTING FOR XTILT, YTILT, AND ZROT.  ALL BASED ON
C     ANGLE IN THE POSITIVE DIRECTION FROM THE POSITIVE
C     X, Y, AND Z AXIS
C
C*************************************************************

          XT=X
          YT=Y*C1-Z*S1
          ZT=Y*S1+Z*C1
          Y=YT
          Z=ZT
          XT=X*C2-Y*S2
          ZT=X*S2+Z*C2
          X=XT
          Z=ZT
          LOCAL(J,1)=X*C3-Y*S3
          LOCAL(J,2)=X*S3+Y*C3
          LOCAL(J,3)=Z

C*************************************************************
C
C     WRITING THE OUTPUT FILE
C
C*************************************************************

   60 CONTINUE
      WRITE(41,230) JNR(K),ZC,K
  230 FORMAT(I10,F10.4,I10)
      DO 65 J=1,200
      WRITE(41,235) (LOCAL(J,N),N=1,3)
  235 FORMAT(3(5X,F10.2))
   65 CONTINUE
      STOP
      END
```

```
C*************************************************************
C
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C     THIS PROGRAM WILL GENERATE A SET OF ANGLES AND TIMES
C     CORRESPONDING TO INPUT FILES CONSISTING OF TORPEDO
C     POSITIONS, AND A CERTAIN DEPTH VELOCITY PROFILE.
C
C     INPUT FILE 43 CONSISTS OF A FILE WITH THE DEPTH AND
C     VELOCITY AT THAT DEPTH.  THE LAST LINE OF THAT FILE
C     CONTAINS THE VALUES FOR V0 AND V1, WHICH ARE THE
C     SINGLE LAYER REGRESSION OF THE DEPTH AND VELOCITY
C
C     INPUT FILE 41 IS THE FILE OF TORPEDO
C     POSITIONS IN THE LOCAL SYSTEM.  THE FIRST LINE OF
C     THIS FILE CONSISTS OF THE SENSOR NUMBER AND DEPTH
C     THIS FILE IS GENERATED BY THE AUTHOR'S "CONVERT"
C     PROGRAM.
C
C     OUTPUT FILE 42 IS THE OUTPUT FILE OF THE TIME AT
C     EACH HYDROPHONE.  EACH LINE REFERS TO A SINGLE
C     POINT COUNT, AND CONSISTS OF THE TIMES AT THE
C     X, Y, Z, AND C HYDROPHONES.  THE FIRST LINE OF
C     THE FILE CONSISTS OF THE SENSOR NUMBER AND THE
C     DEPTH OF THE SENSOR.
C
C     MAJOR VARIABLES:
C     DEPTH -- AN ARRAY CONTAINING THE DEPTH IN THE LAYERS
C     VEL --   AN ARRAY CONTAINING THE VELOCITY IN THE LAYERS
C     L --   AN ARRAY CONTAINING THE LAYER MIDPOINT VELOCITY
C     G --   AN ARRAY CONTAINING THE LAYER GRADIENTS
C     ACX,ACY,ACZ -- THE X,Y, AND X COORDINATES OF THE SENSOR
C              IN THE LOCAL COORDINATE SYSTEM.
C     DP --    THE DISTANCE BETWEEN HYDROPHONES OF THE SENSOR
C     POSIT -- A TWO DIMENSIONAL ARRAY WHERE THE FIRST
C              SUBSCRIPT REFERS TO THE POINT COUNT, AND THE
C              SECOND SUBSCRIPT REFERS TO THE X,Y,AND Z VALUE
C              FOR THE POSITION OF THAT POINT COUNT IN THE
C              LOCAL COORDINATE SYSTEM.
C     TX,TY,TZ -- THE X,Y, AND Z COORDINATES OF THE TARGET
C     RESULT -- A TWO DIMENSIONAL ARRAY WHERE THE FIRST
C               SUBSCRIPT REFERS TO THE POINT COUNT, AND THE
C               SECOND SUBSCRIPT REFERS TO HYDROPHONE NUMBER
C               X,Y,Z AND C.  IT CONTAINS THE GENERATED TIMES
C               FOR THAT HYDROPHONE, AT THAT POINT COUNT
C
C*************************************************************

      PROGRAM MAKE
      DIMENSION DEPTH(53),VEL(53),POSIT(200,3),RESULT(200,4)
      DIMENSION S(4,3), L(53), S(4,3)
      REAL*8 V0,V1,TIME,ANGLE,R,S,L,G,DL
```

```
      REAL*8 DEPTH,VEL,D,POSIT,RESULT,Z

C*************************************************************
C     READING IN THE DEPTH AND VELOCITY PROFILE
C*************************************************************

      WRITE(*,100)
  100 FORMAT(/,5X,'READING IN THE DEPTH VELOCITY PROFILE')
      DO 5 J=1,53
      READ(43,11) DEPTH(J),VEL(J)
    5 CONTINUE
   11 FORMAT(2F10.4)
      READ(43,11) V0,V1

C*************************************************************
C     SETTING UP LAYER END POINTS AND THE GRADIENT VALUES
C*************************************************************

      DL=25.0D0
      DO 10 J=1,52
      L(J)=DEPTH(J)-(DL/2.0D0)
      G(J)=(VEL(J+1)-VEL(J))/DL
   10 CONTINUE
      L(53)=L(52)+DL
      G(53)=G(52)+((G(52)+G(44))/8.0D0)

C*************************************************************
C     READING THE POSITION OF THE TORPEDO
C*************************************************************

      READ(41,210) NR,D,T
  210 FORMAT(I10,F10.4,I10)
      DO 20 J=1,200
      READ (41,200) (POSIT(J,K),K=1,3)
  200 FORMAT(3(5X,F10.2))
   20 CONTINUE

C*************************************************************
C     CALCULATION OF THE SENSOR POSITION
C*************************************************************

  212 DO 30 J=1,4
      S(J,1)=-15.0D0
      S(J,2)=-15.0D0
      S(J,3)=D+15.0D0
  30  CONTINUE
      S(1,1)=15.0D0
      S(2,2)=15.0D0
      S(3,3)=D-15.0D0
```

```
C*************************************************************
C     CALCULATE THE TIME AND ANGLE FOR EACH POSITION
C*************************************************************

      DO 300 J=1,200
      DO 300 K=1,4
      R= (POSIT(J,1)-S(K,1))*(POSIT(J,1)-S(K,1))
      R=R+((POSIT(J,2)-S(K,2))*(POSIT(J,2)-S(K,2)))
      R=DSQRT(R)
      Z=S(K,3)
      CALL TGEN(L,G,VEL,Z,R,POSIT(J,3),
     1ANGLE,TIME,V0,V1,DEPTH)
      RESULT(J,K)=TIME
  300 CONTINUE

C*************************************************************
C     WRITE OUTPUT FILE
C*************************************************************

      WRITE(42,210) NR,D,T
      DO 400 J=1,200
      WRITE(42,230) (RESULT(J,K),K=1,4)
  230 FORMAT(4F20.15)
  400 CONTINUE
      STOP
      END
```

```
C*************************************************************
C
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C     THIS PROGRAM TAKES A FILE OF TIMES (GENERATED BY
C     THE AUTHOR'S "MAKE" PROGRAM), AND CALCULATES
C     THE POSITION OF THE TORPEDO IN THE LOCAL COORDINATE
C     SYSTEM.
C
C     INPUT FILE 43 CONSISTS OF A FILE WITH THE DEPTH AND
C     VELOCITY AT THAT DEPTH.  THE LAST LINE OF THAT FILE
C     CONTAINS THE VALUES FOR V0 AND V1, WHICH ARE THE
C     SINGLE LAYER REGRESSION OF THE DEPTH AND VELOCITY
C
C     INPUT FILE 42 IS THE FILE OF THE TIME AT
C     EACH HYDROPHONE.  EACH LINE REFERS TO A SINGLE
C     POINT COUNT, AND CONSISTS OF THE TIMES AT THE
C     X, Y, Z, AND C HYDROPHONES.  THE FIRST LINE OF
C     THE FILE CONSISTS OF THE SENSOR NUMBER AND THE
C     DEPTH OF THE SENSOR.
C
C     OUTPUT FILE 48 CONSISTS OF THE OUTPUT FILE OF
C     TORPEDO POSITIONS IN THE LOCAL SYSTEM.
C     THE FIRST LINE OF THIS FILE CONSISTS OF THE
C     SENSOR NUMBER AND DEPTH
C
C     MAJOR VARIABLES:
C     DEPTH -- AN ARRAY CONTAINING THE DEPTH IN THE LAYERS
C     VEL --   AN ARRAY CONTAINING THE VELOCITY IN THE LAYERS
C     L --   AN ARRAY CONTAINING THE LAYER MIDPOINT VELOCITY
C     G --   AN ARRAY CONTAINING THE LAYER GRADIENTS
C     POSIT -- A TWO DIMENSIONAL ARRAY WHERE THE FIRST
C              SUBSCRIPT REFERS TO THE POINT COUNT, AND THE
C              SECOND SUBSCRIPT REFERS TO THE X,Y,AND Z VALUE
C              FOR THE POSITION OF THAT POINT COUNT IN THE
C              LOCAL COORDINATE SYSTEM.
C     TX,TY,TZ -- THE X,Y, AND Z COORDINATES OF THE TARGET
C     RESULT -- A TWO DIMENSIONAL ARRAY WHERE THE FIRST
C               SUBSCRIPT REFERS TO THE POINT COUNT, AND THE
C               SECOND SUBSCRIPT REFERS TO HYDROPHONE NUMBER
C               X,Y,Z AND C.  IT CONTAINS THE GENERATED TIMES
C               FOR THAT HYDROPHONE, AT THAT POINT COUNT
C     ALL REAL NUMBERS ARE REAL*8.  ALL ANGLES IN RADIANS
C
C*************************************************************

      PROGRAM NAVY
      DIMENSION DEPTH(53),VEL(53),POSIT(200,3),L(53),G(53)
      DIMENSION RESULT(200,4)
      REAL*8 V0,V1,TIME,ANGLE,R,DL,V2,V3,DPH,RNG,L
      REAL*8 DEPTH,VEL,D,POSIT,RESULT,VXYC,VZ
      REAL*8 XTX2,XTY2,XTZ2,XTC2,XA,YA,ZA,RC,R2,G
```

```
C*************************************************************
C
C     READING IN THE DEPTH AND VELOCITY PROFILE
C
C*************************************************************


      WRITE(*,100)
  100 FORMAT(/,5X,'READING IN THE DEPTH VELOCITY PROFILE')
      DO 5 J=1,53
      READ(43,11) DEPTH(J),VEL(J)
    5 CONTINUE
   11 FORMAT(2F10.4)
      READ(43,11) V0,V1

C*************************************************************
C
C     SETTING UP LAYER END POINTS AND THE GRADIENT VALUES
C
C*************************************************************

      DL=25.0D0
      DO 10 J=1,52
      L(J)=DEPTH(J)-(DL/2.0D0)
      G(J)=(VEL(J+1)-VEL(J))/DL
   10 CONTINUE
      L(53)=L(52)+DL
      G(53)=G(52)+((G(52)+G(44))/8.0D0)


C*************************************************************
C
C     ENTERING THE POSITION OF THE TORPEDO
C
C*************************************************************

      READ(42,600) NR,D,JT
  600 FORMAT(I10,F10.4,I10)
      DO 605 J=1,200
      READ(42,602,END=601) (RESULT(J,K),K=1,4)
  602 FORMAT(4F20.15)
  605 CONTINUE

C*************************************************************
C
C     CALCUALTING THE VELOCITY AT SENSOR DEPTH
C
C*************************************************************

  601 K=-1
```

```
      KL=-1
      DO 610 J=1,53
      IF((D+15.0D0).LT.DEPTH(J)) K=J
      IF((D-15.0D0).LT.DEPTH(J)) KL=J
  610 CONTINUE
      VXYC=VEL(K)+(G(K)*(D+15.0D0-DEPTH(K)))
      VZ = VEL(KL)+(G(KL)*(D-15.0D0-DEPTH(KL)))
      V2=VXYC*VXYC
      V3=VZ*VZ
      DO 700 J=1,200

C*************************************************************
C
C     CALCULATION OF APPARANT POSITION USING NAVY METHOD
C
C*************************************************************

      XTX2=RESULT(J,1)*RESULT(J,1)
      XTY2=RESULT(J,2)*RESULT(J,2)
      XTZ2=RESULT(J,3)*RESULT(J,3)
      XTC2=RESULT(J,4)*RESULT(J,4)
      XA=V2*(XTC2-XTX2)/60.0D0
      YA=V2*(XTC2-XTY2)/60.0D0
      ZA=V3*(XTC2-XTZ2)/60.0D0
      R=DSQRT((XA*XA)+(YA*YA)+(ZA*ZA))
      ANGLE=DASIN(ZA/R)
      RC=((XA+15.0D0)*(XA+15.0D0))+((YA+15.0D0)*(YA+15.0D0))
      RC=RC+((ZA+15.0D0)*(ZA+15.0D0))
      RC=DSQRT(RC)
      TIME=RESULT(J,4)*R/RC

C*************************************************************
C
C     ISOGRADIENT RAYTRACING USING THE TIME, AND ANGLE
C     CALCUATED BY THE NAVY METHOD.
C
C*************************************************************

      CALL GTRC(TIME,ANGLE,D,VEL,DPH,RNG,L,G)

C*************************************************************
C
C     REFINING THE APPARANT POSITION USING THE RAYTRACED
C     DEPTH AND RANGE
C
C*************************************************************

      R2=DSQRT((XA*XA)+(YA*YA))
      POSIT(J,1)=RNG*XA/R2
      POSIT(J,2)=RNG*YA/R2
      POSIT(J,3)=DPH
```

```
  700 CONTINUE

C*************************************************************
C     WRITING THE OUTPUT FILE
C*************************************************************

      WRITE(48,600) NR,D,JT
      DO 715 J=1,200
      WRITE(48,710) (POSIT(J,K),K=1,3)
  710 FORMAT(10X,3(5X,F10.2))
  715 CONTINUE
      STOP
      END
```

```
C*************************************************************
C     PROGRAMMER:  WILLIAM M. KROSHL
C
C     THIS PROGRAM CONVERTS A USER DESIGNATED INPUT FILE IN
C     LOCAL RANGE COORDINATES TO A FILE IN GLOBAL COORDINATES
C     FOR ANY DESIRED ARRAY.
C
C     INPUT FILE 40 IS THE INPUT FILE FOR ALL SENSORS.  EACH
C     LINE OF THE INPUT FILE CONSISTS OF THE SENSOR NUMBER,
C     THE DOWNRANGE, CROSSRANGE, AND DEPTH OF THE SENSOR,
C     AND THE VALUES OF X-TILT, Y-TILT, AND Z-ROT.
C
C     INPUT FILE 48 IS THE INPUT FILE OF LOCAL POSITIONS
C     THE FIRST LINE WILL BE THE SENSOR NUMBER, SENSOR
C     DEPTH, AND INDEX NUMBER (WHICH CAN BE IGNORED)
C     EACH SUBSEQUENT LINE IS THE DOWNRANGE, CROSSRANGE, AND
C     DEPTH COORDINATE FOR THE TRACK IN THE LOCAL SYSTEM.
C
C     OUTPUT FILE 49 IS THE GLOBAL POSITION OF THE TORPEDO.
C     EACH LINE OF THE INPUT FILE CONSISTS OF THE DOWNRANGE,
C     CROSSRANGE, AND DEPTH, ONE LINE PER POINT COUNT
C
C     ALL CALCULATIONS ARE MADE USING REAL*4 VARIABLES.
C
C*************************************************************

      PROGRAM UNCVRT
      DIMENSION LOCAL(200,3), GLOBAL(200,3), POSIT(50,6)
      DIMENSION JNR(50)
      REAL*4 LOCAL

C*************************************************************
C
C     READ INPUT FILES
C
C*************************************************************

      DO 10 J=1,50
         READ(40,205,END=5) JNR(J),(POSIT(J,K),K=1,6)
  205    FORMAT(I2,6F10.4)
   10 CONTINUE
    5 READ(48,306) I,Z,JT
  306 FORMAT(I10,F10.4,I10)
      DO 20 J=1,200
         READ(48,210,END=50) (GLOBAL(J,K),K=1,3)
  210    FORMAT(10X,3(5X,F10.2))
   20 CONTINUE
   50 K=-1
      DO 55 J=1,50
         IF(JNR(J).EQ.I) K=J
   55 CONTINUE
      IF(K.LT.0) GOTO 50
```

```
C*************************************************************
C     DETERMINING THE PROPER VALUES FOR THE CONVERSION
C*************************************************************

      XC=POSIT(K,1)*(-1.0)
      YC=POSIT(K,2)*(-1.0)
      ZC=POSIT(K,3)*(-1.0)
      XTILT=POSIT(K,4)*(-1.0)
      YTILT=POSIT(K,5)*(-1.0)
      ZROT=POSIT(K,6)*(-1.0)
      C1=COS(YTILT)
      S1=SIN(YTILT)
      C2=COS(XTILT)
      S2=SIN(XTILT)
      C3=COS(ZROT)
      S3=SIN(ZROT)

      DO 60 J=1,200

C*************************************************************
C     CONVERTING FOR LATERAL DISPLACEMENT
C*************************************************************

         X=GLOBAL(J,1)-XC
         Y=GLOBAL(J,2)-YC
         Z=GLOBAL(J,3)

C*************************************************************
C
C     CONVERTING FOR XTILT, YTILT, AND ZROT.  ALL BASED
C     ON ANGLE IN THE POSITIVE DIRECTION FROM THE
C     POSITIVE AXIS
C
C*************************************************************

         XT=X
         YT=Y*C1-Z*S1
         ZT=Y*S1+Z*C1
         Y=YT
         Z=ZT
         XT=X*C2-Y*S2
         ZT=X*S2+Z*C2
         X=XT
         Z=ZT
         LOCAL(J,1)=X*C3-Y*S3
         LOCAL(J,2)=X*S3+Y*C3
         LOCAL(J,3)=Z
   60 CONTINUE
```

```
C*************************************************************
C
C     WRITING THE OUTPUT FILE
C
C*************************************************************

      WRITE(49,230) JNR(K),Z,JT
  230 FORMAT(I10,F10.4,I10)
      DO 65 J=1,200
      WRITE(49,235) (LOCAL(J,K),K=1,3)
  235 FORMAT(3(5X,F10.2))
   65 CONTINUE
      STOP
      END
```

APPENDIX D

Sample Depth Velocity Profile

Depth	Velocity
25.0000	4879.1566
50.0000	4871.7664
75.0000	4849.6522
100.0000	4845.7903
125.0000	4845.4850
150.0000	4845.6796
175.0000	4846.0873
200.0000	4846.5014
225.0000	4847.3039
250.0000	4847.8543
275.0000	4847.7813
300.0000	4848.0356
325.0000	4848.5005
350.0000	4849.1797
375.0000	4850.7174
400.0000	4851.7636
425.0000	4852.8447
450.0000	4853.6115
475.0000	4854.2449
500.0000	4855.0154
525.0000	4855.4043
550.0000	4856.3507
575.0000	4857.7940
600.0000	4859.0185
625.0000	4859.9290
650.0000	4861.7371
675.0000	4862.8009
700.0000	4863.4343
725.0000	4864.7141
750.0000	4865.8117
775.0000	4867.6003
800.0000	4868.9115
825.0000	4869.6331
850.0000	4870.6119
875.0000	4871.2721
900.0000	4872.7002
925.0000	4873.5612
950.0000	4874.0619
975.0000	4874.8938
1000.0000	4875.5581
1025.0000	4876.0666
1050.0000	4876.6192
1075.0000	4877.2783
1100.0000	4877.9348

APPENDIX D (CONTINUED)

Depth	Velocity
1125.0000	4878.7165
1150.0000	4879.4371
1175.0000	4879.9722
1200.0000	4880.5158
1225.0000	4880.9781
1250.0000	4881.5758
1275.0000	4882.3175
1300.0000	4882.8900
1325.0000	4883.4339

APPENDIX E

Nanoose Range: Sensor Positions
All Positions in Global Coordinates

Sensor	Downrange	Crossrang	Depth
0	12188.0	-131.5	-1295.3
1	19463.1	-174.9	-1308.7
2	26991.3	-109.8	-1323.2
3	34436.0	-334.6	-1325.0
4	42031.3	-99.6	-1320.8
5	49497.0	-25.2	-1315.5
6	58972.7	-21.2	-1308.5
7	64680.6	15.3	-1353.3
8	71995.2	34.9	-1303.8
9	3.0	3.0	1.0
10	47100.0	-3600.0	-1300.0
11	23173.8	-6488.4	-1312.0
12	30731.2	-6553.0	-1312.9
13	38213.6	-6640.7	-1323.0
14	45647.0	-6513.1	-1324.7
15	53249.4	-6354.8	-1316.6
16	60848.5	-6374.8	-1316.1
17	68210.3	-6515.2	-1318.6
54	38010.2	5467.5	-1215.0
55	45645.7	6369.6	-1188.1
56	53180.1	6417.9	-1718.8
57	60745.7	6419.4	-1088.2
23	41605.1	-12150.1	-1268.2
24	49456.4	-12973.5	-1307.6
25	56973.1	-13003.7	-1210.4
26	54358.4	-12976.0	-1256.4
27	22119.6	-15908.7	83.0
28	45000.0	1500.0	-1350.0

APPENDIX F

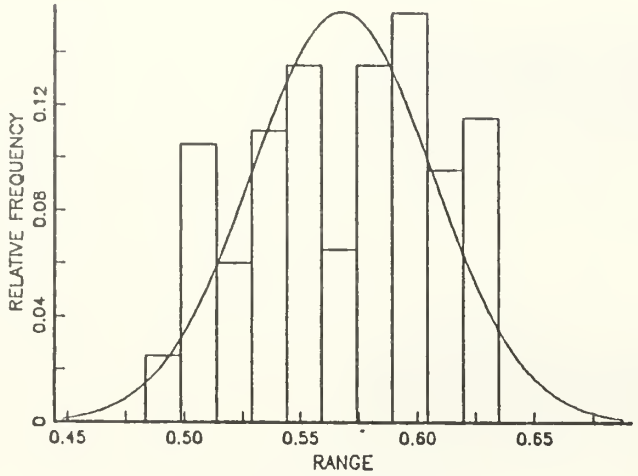
This appendix contains the graphs for the Navy Base Case model (Case I). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

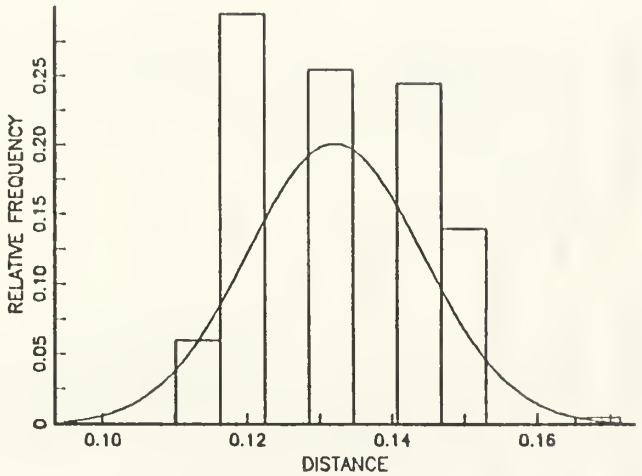
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

BASE CASE: NAVY METHOD

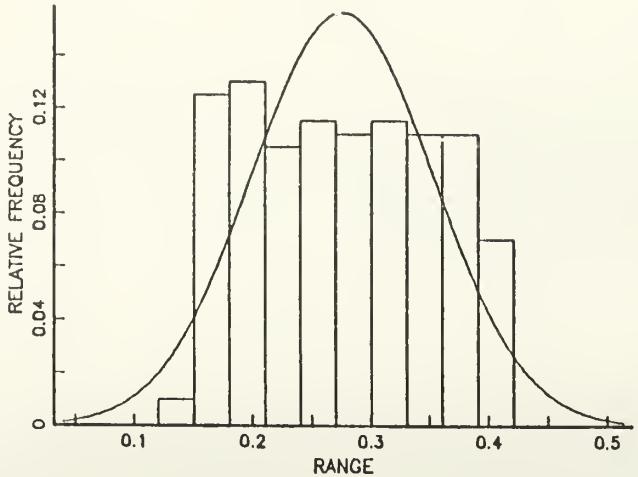
TRACK 1 SENSOR 54



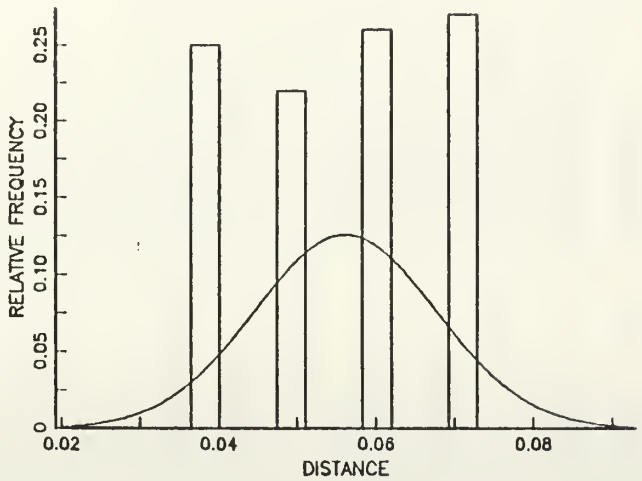
TRACK 1 SENSOR 54



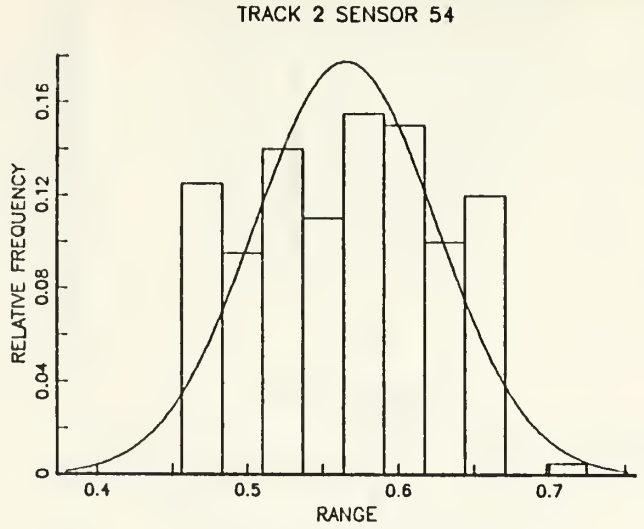
TRACK 1 SENSOR 55

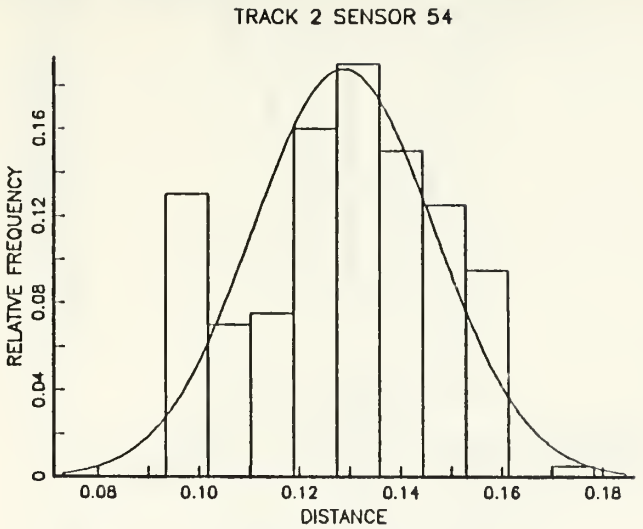


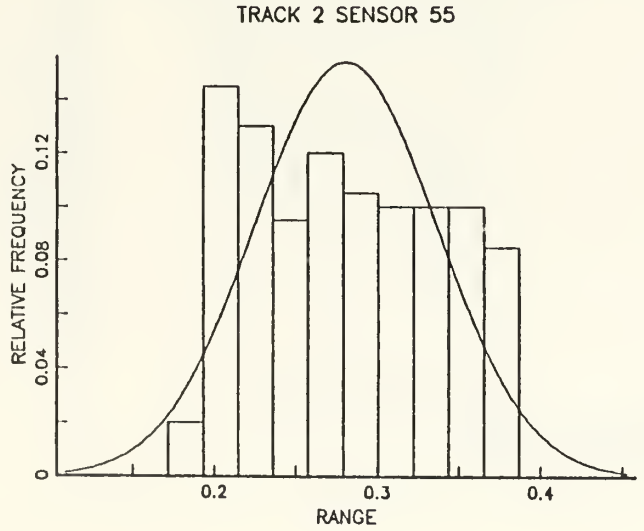
TRACK 1 SENSOR 55

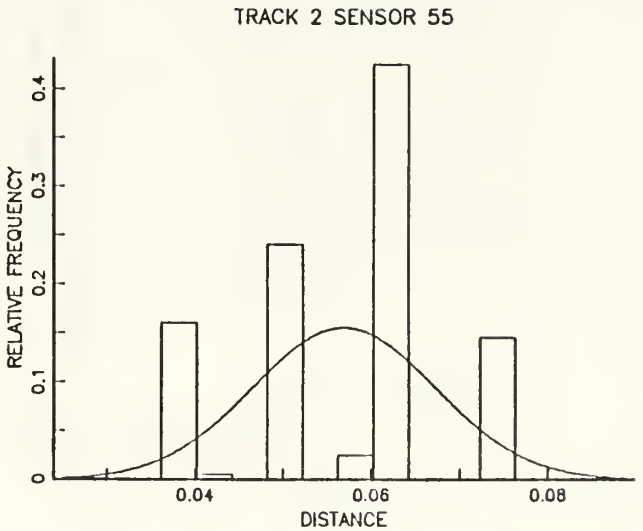


BASE CASE: NAVY METHOD



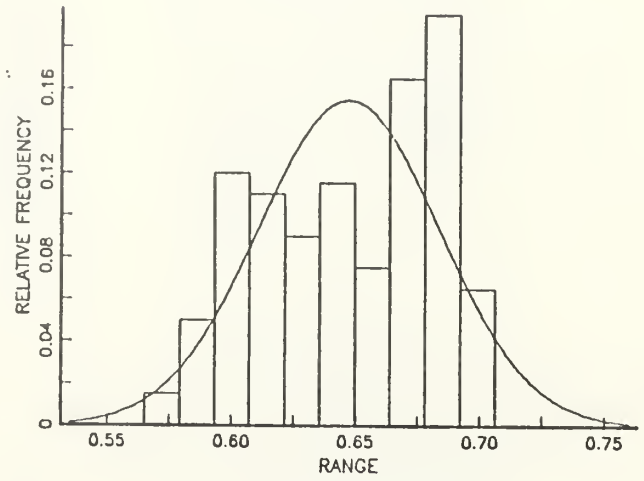




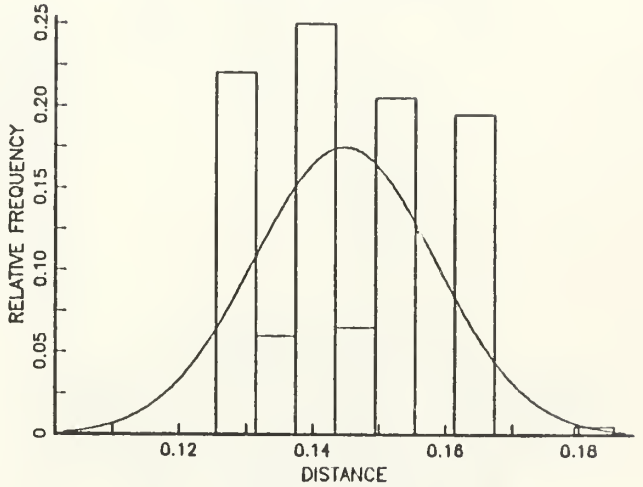


BASE CASE: NAVY METHOD

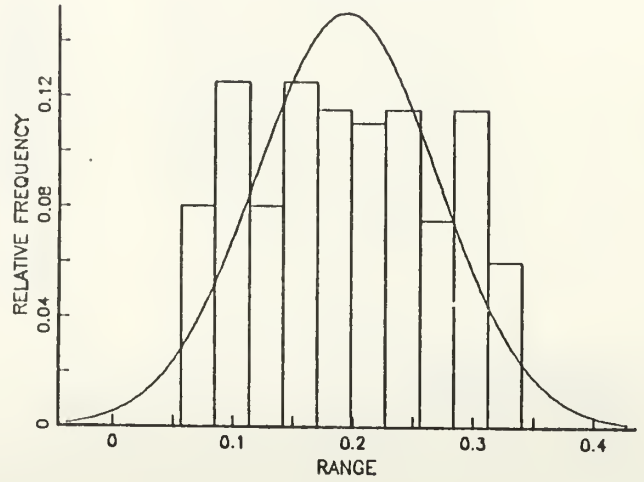
TRACK 3 SENSOR 54



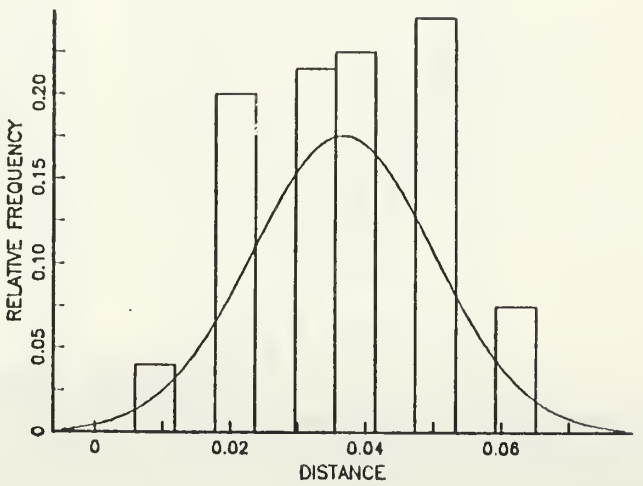
TRACK 3 SENSOR 54



TRACK 3 SENSOR 55

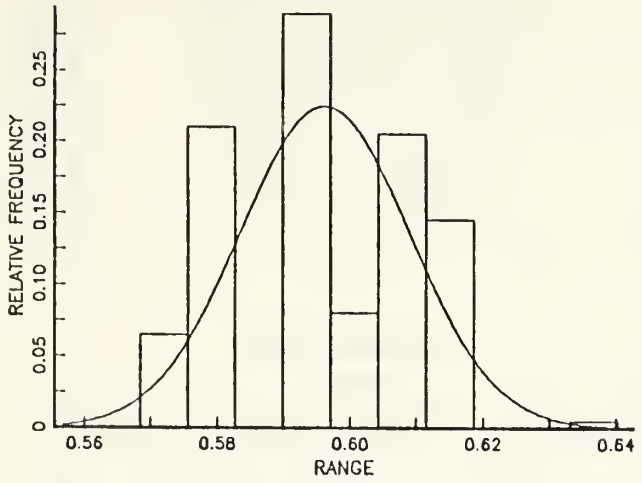


TRACK 3 SENSOR 55

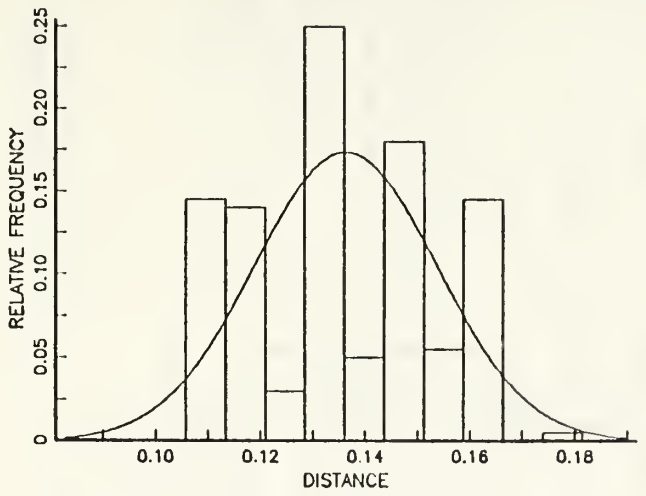


BASE CASE: NAVY METHOD

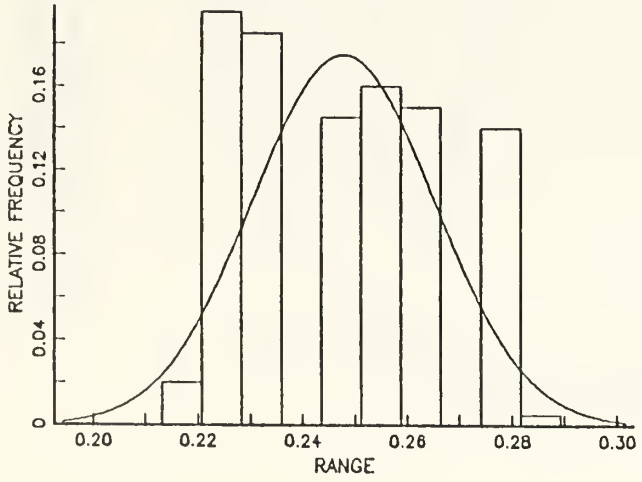
TRACK 4 SENSOR 54



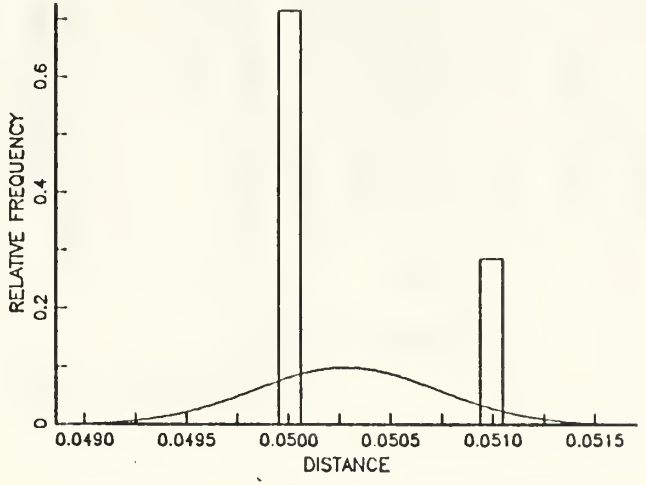
TRACK 4 SENSOR 54



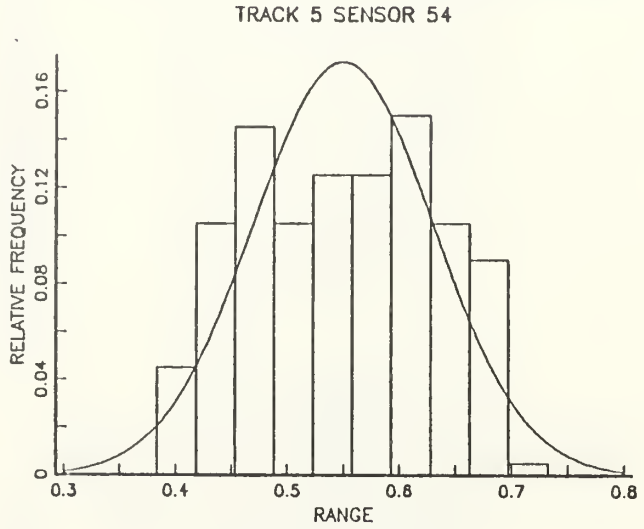
TRACK 4 SENSOR 55

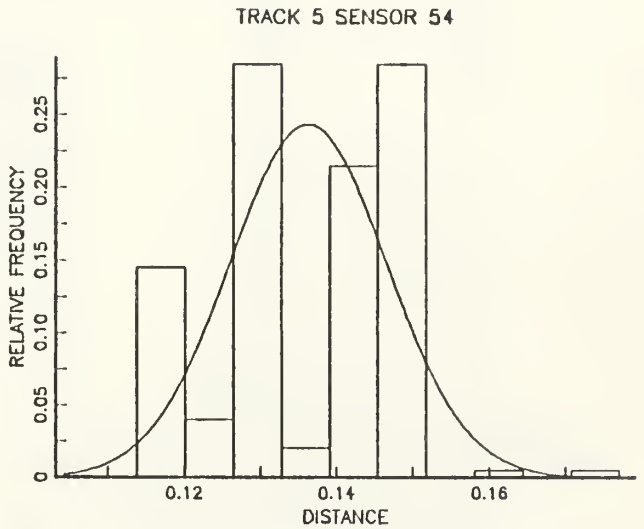


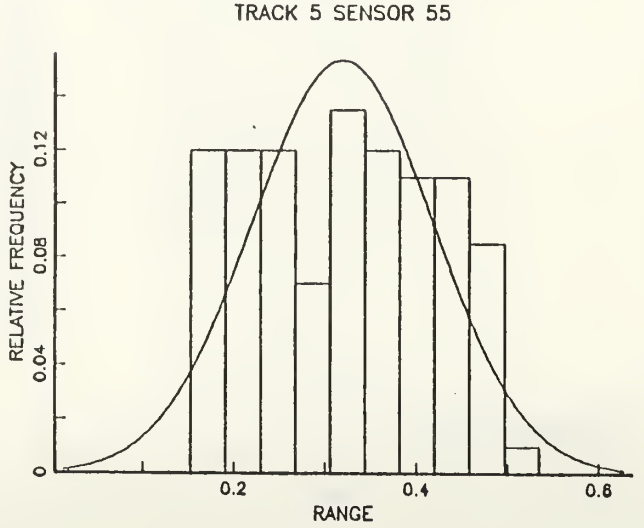
TRACK 4 SENSOR 55

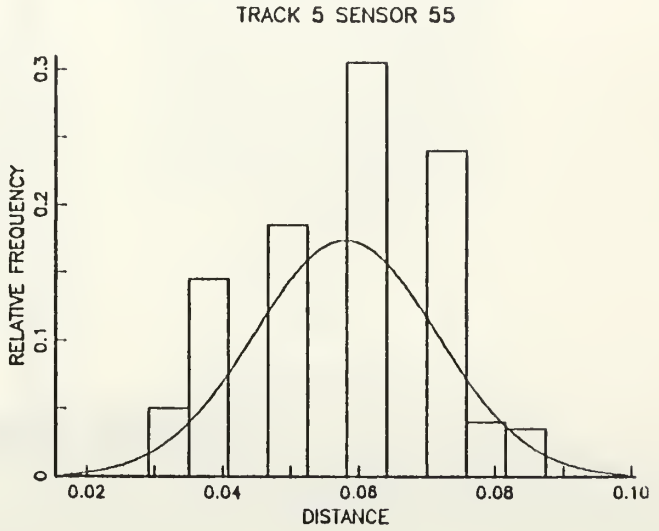


BASE CASE: NAVY METHOD









APPENDIX G

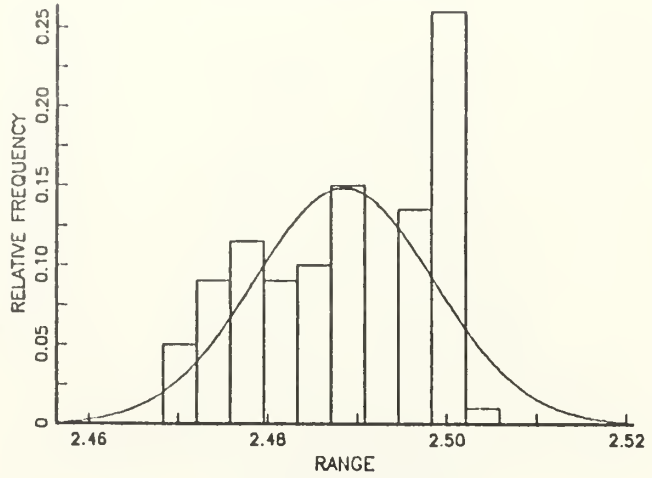
This appendix contains the graphs for the Constant Time Error model (Case II). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

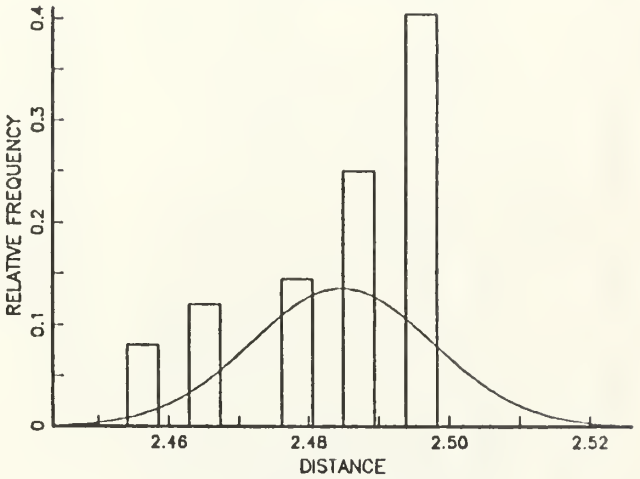
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

CONSTANT ERROR: PLUS 0.0005

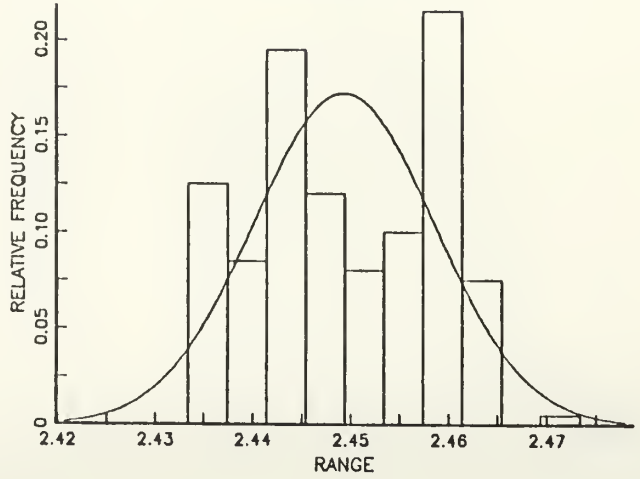
TRACK 1 SENSOR 54



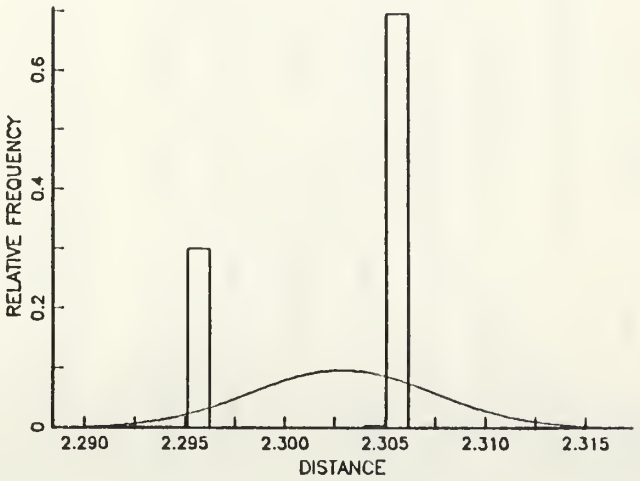
TRACK 1 SENSOR 54



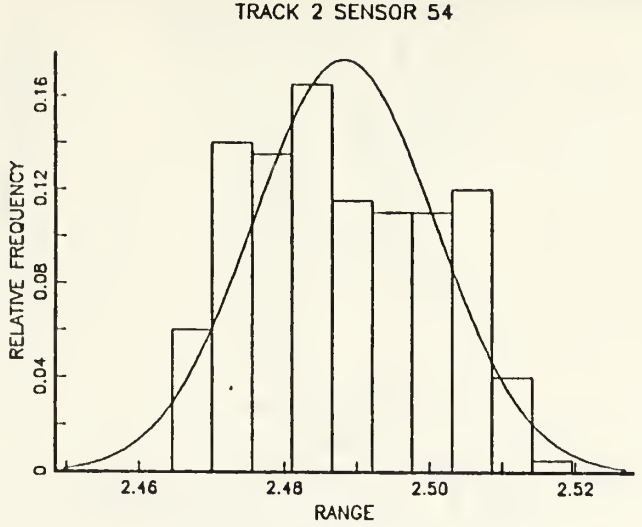
TRACK 1 SENSOR 55

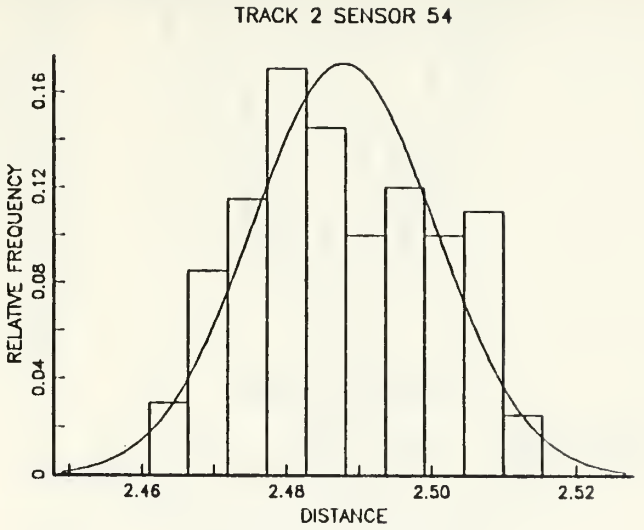


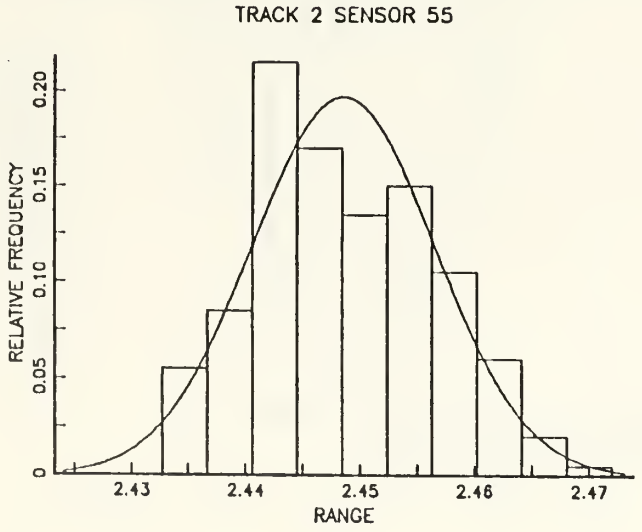
TRACK 1 SENSOR 55

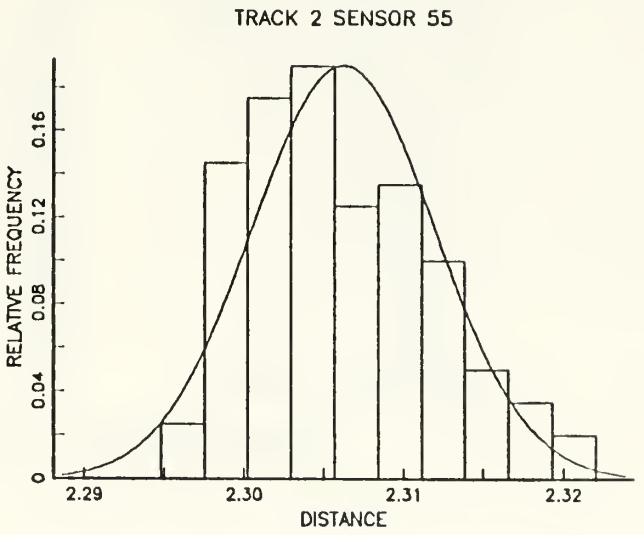


CONSTANT ERROR: PLUS 0.0005



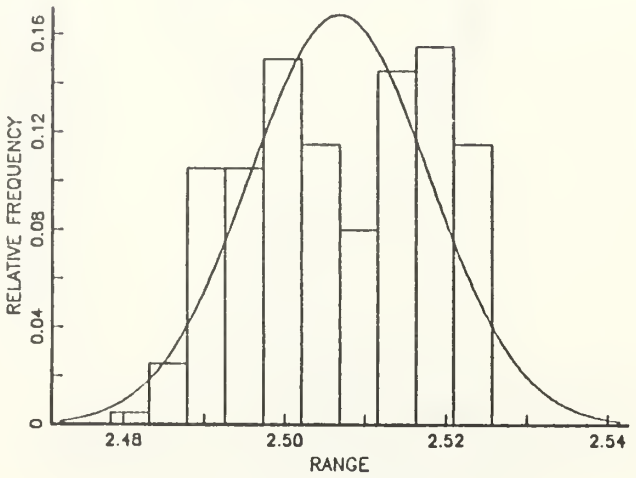




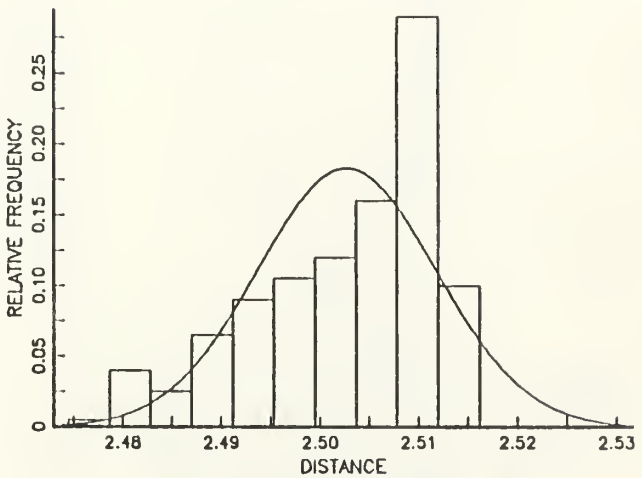


CONSTANT ERROR: PLUS 0.0005

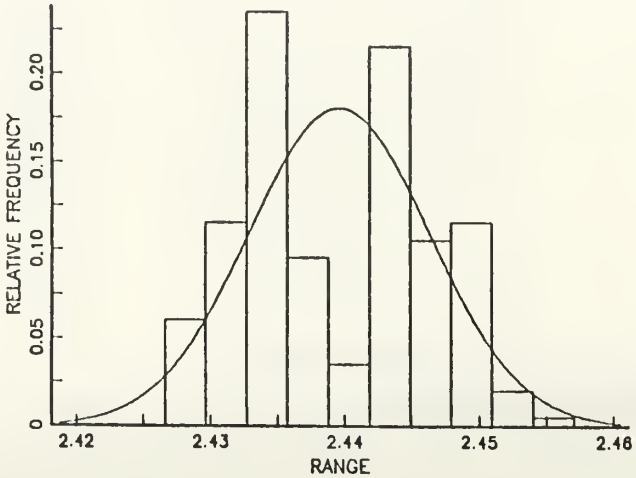
TRACK 3 SENSOR 54



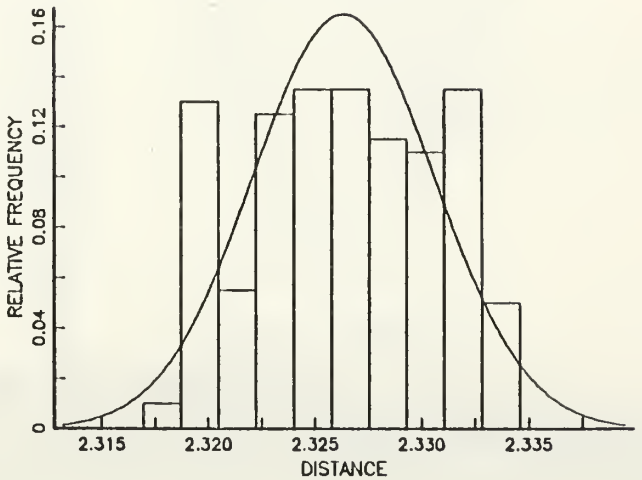
TRACK 3 SENSOR 54



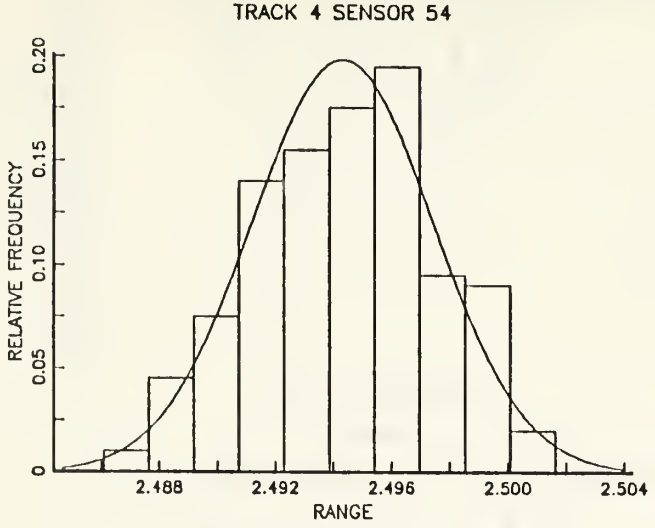
TRACK 3 SENSOR 55

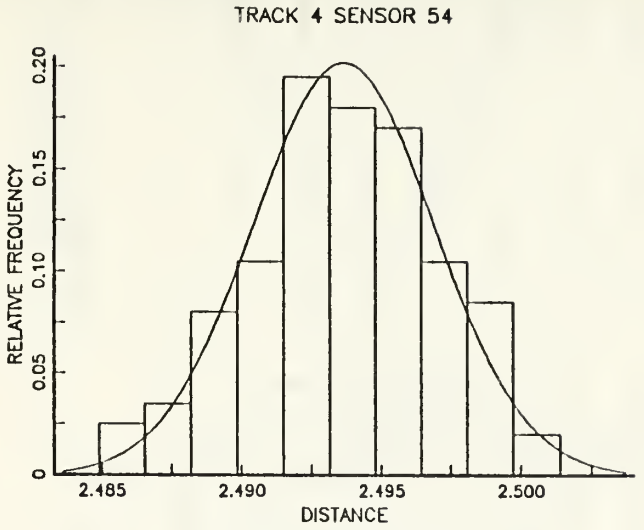


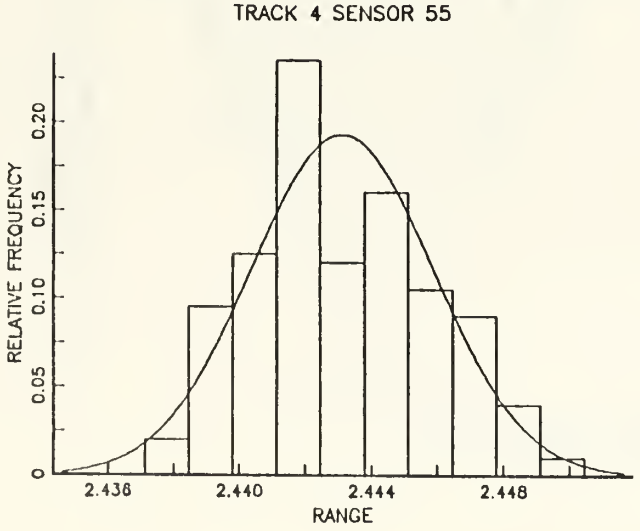
TRACK 3 SENSOR 55

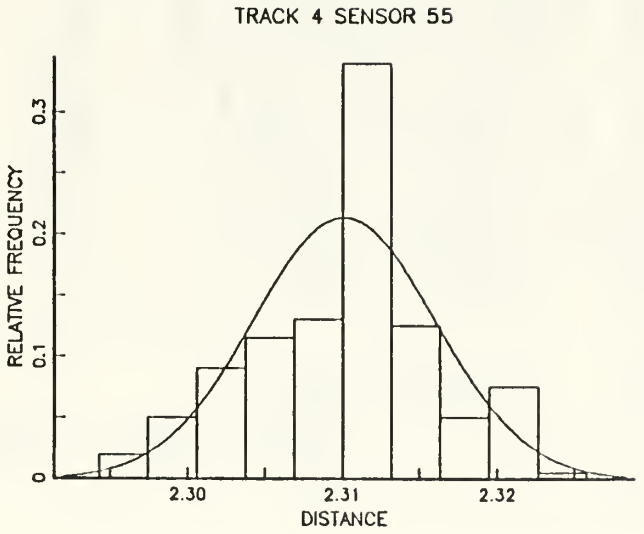


CONSTANT ERROR: PLUS 0.0005



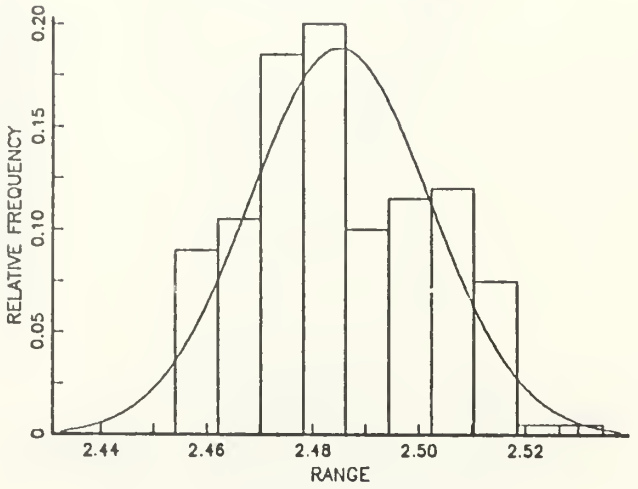




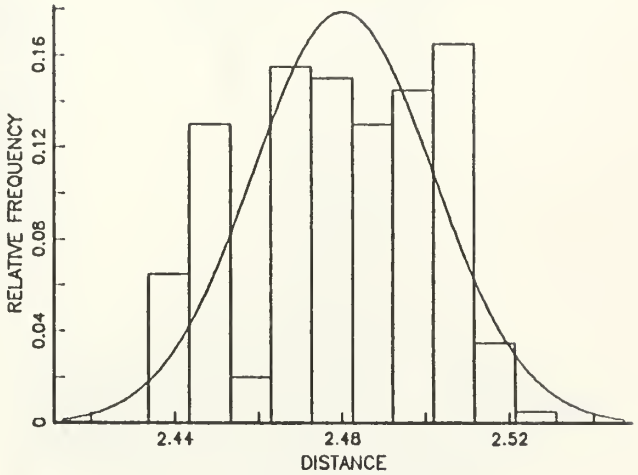


CONSTANT ERROR: PLUS 0.0005

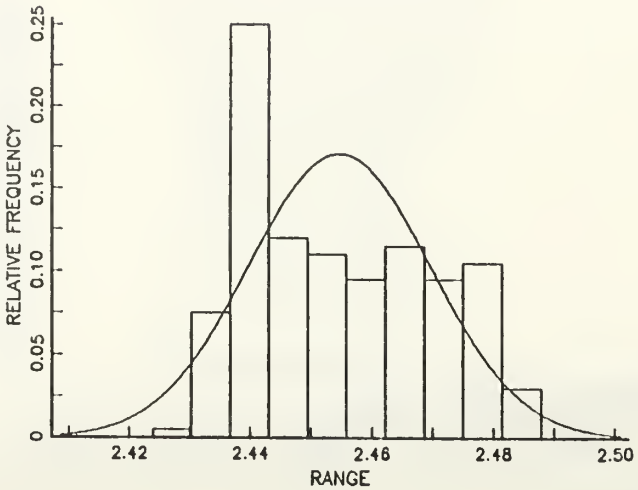
TRACK 5 SENSOR 54



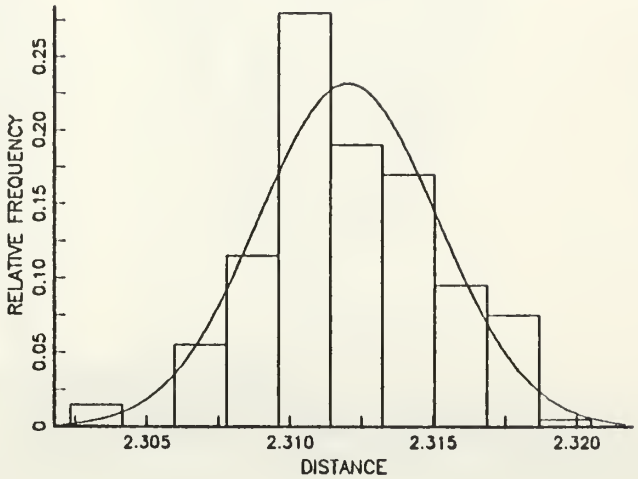
TRACK 5 SENSOR 54



TRACK 5 SENSOR 55



TRACK 5 SENSOR 55



APPENDIX H

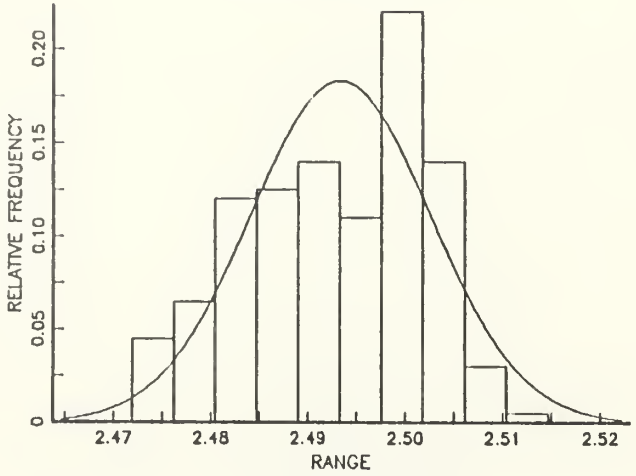
This appendix contains the graphs for the Constant Time Error model (Case III). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

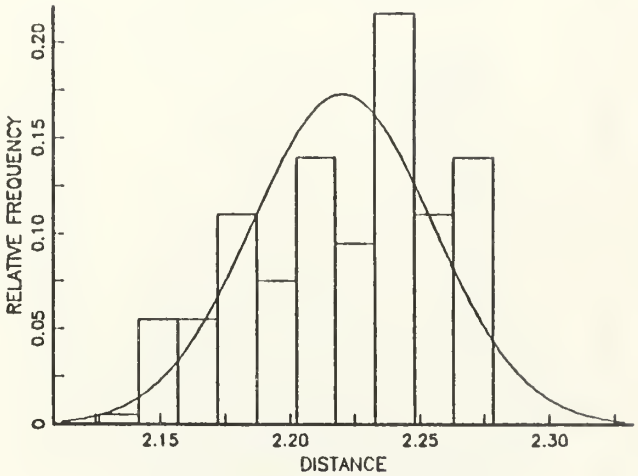
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

CONSTANT ERROR: MINUS 0.0005

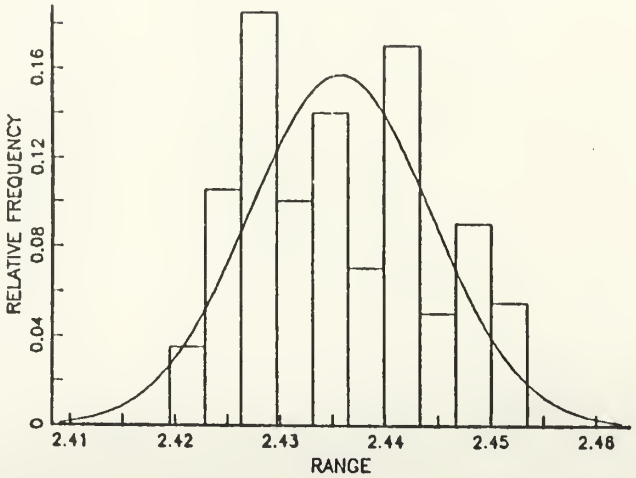
TRACK 1 SENSOR 54



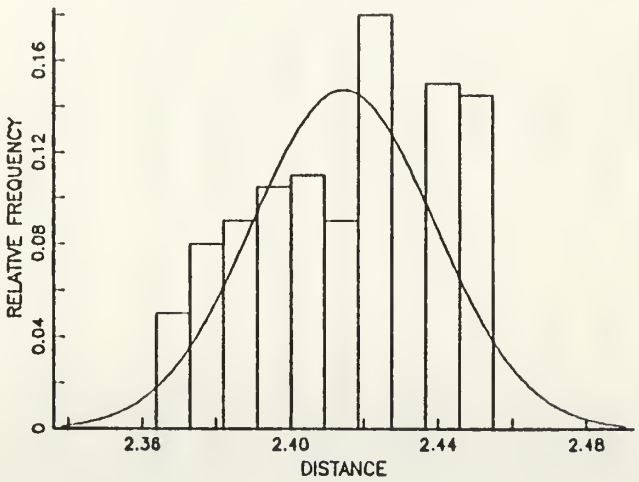
TRACK 1 SENSOR 54



TRACK 1 SENSOR 55

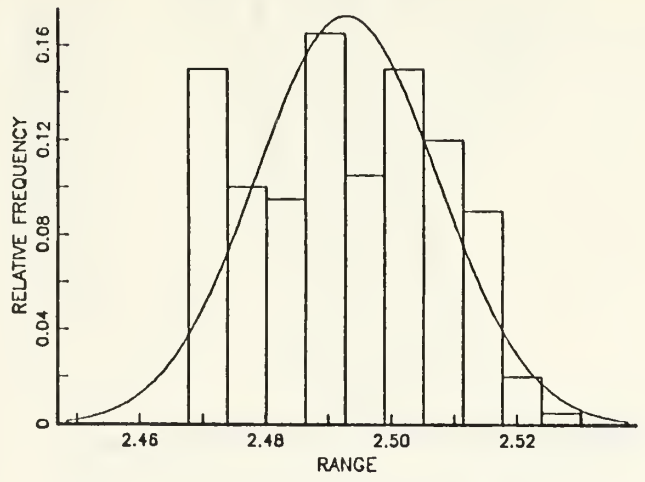


TRACK 1 SENSOR 55

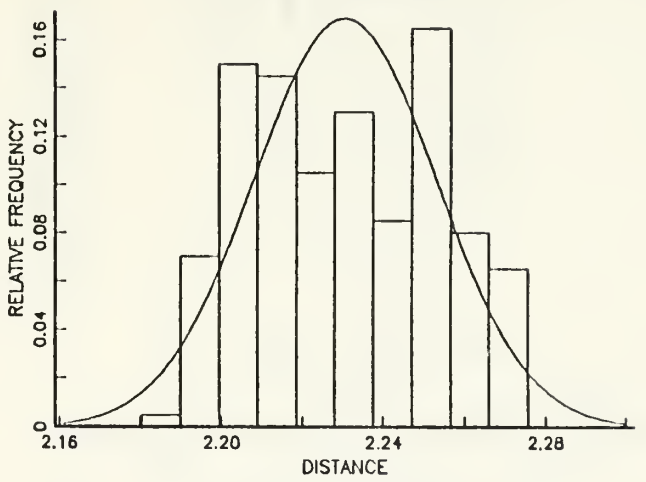


CONSTANT ERROR: MINUS 0.0005

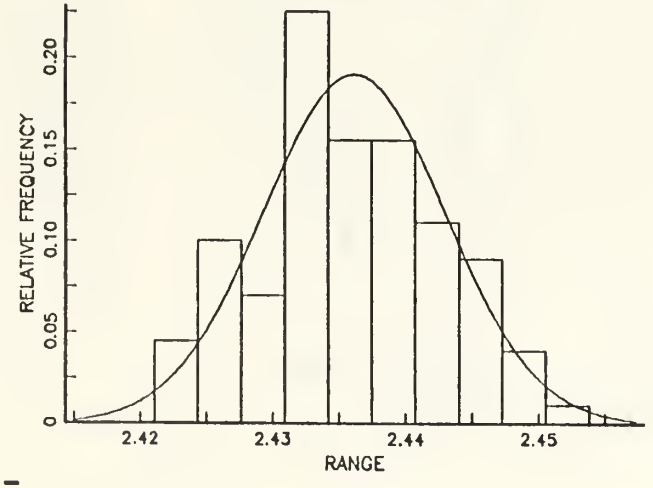
TRACK 2 SENSOR 54



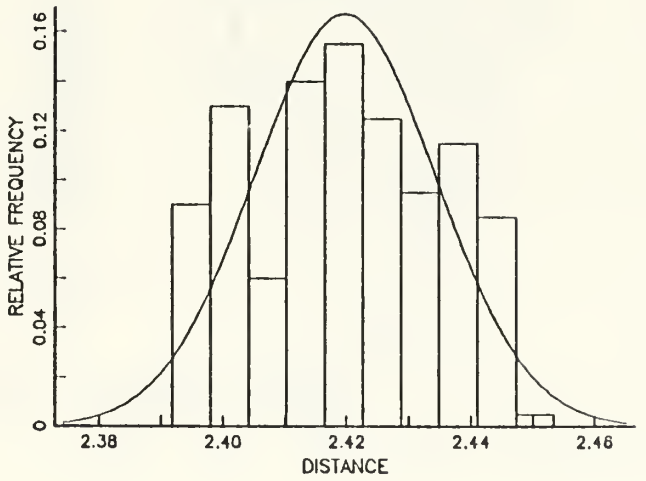
TRACK 2 SENSOR 54



TRACK 2 SENSOR 55

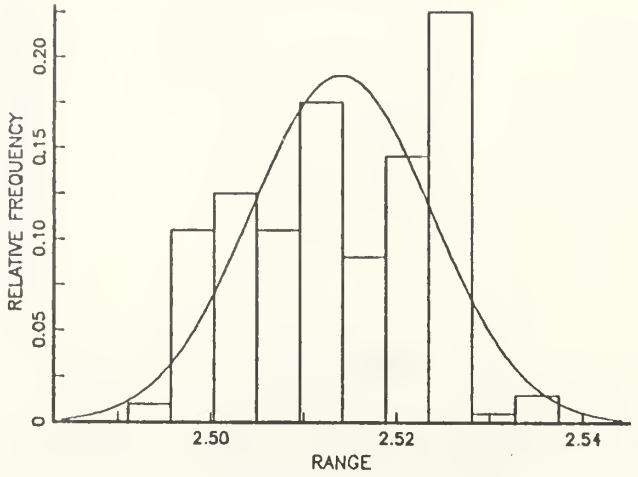


TRACK 2 SENSOR 55

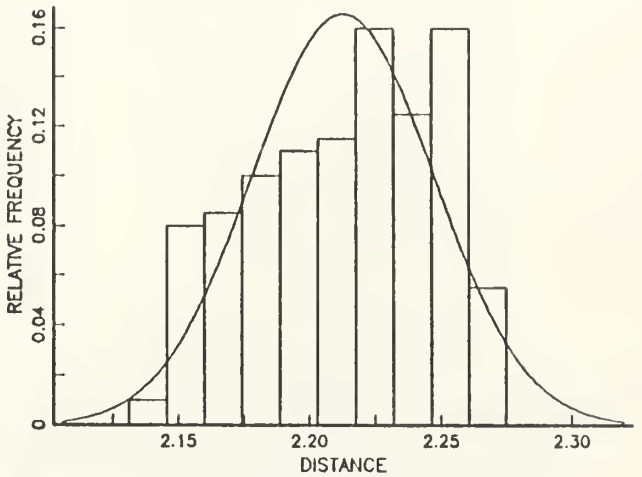


CONSTANT ERROR: MINUS 0.0005

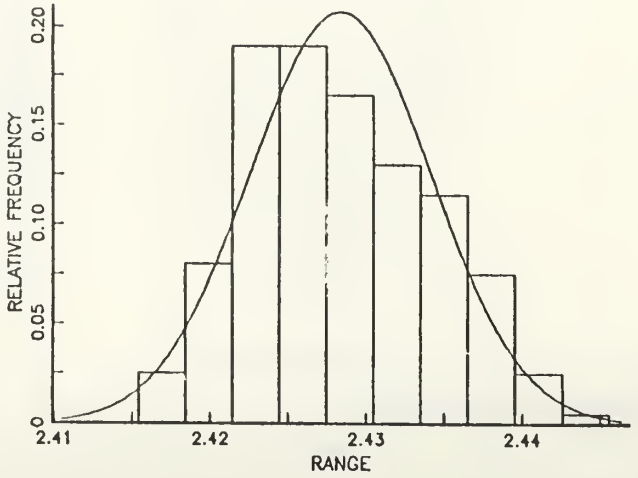
TRACK 3 SENSOR 54



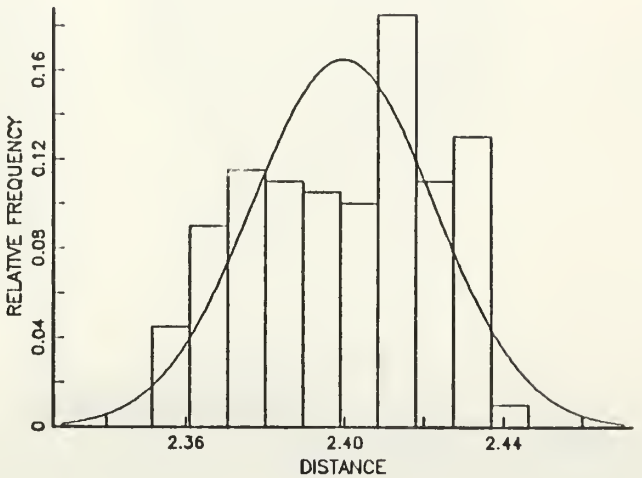
TRACK 3 SENSOR 54



TRACK 3 SENSOR 55

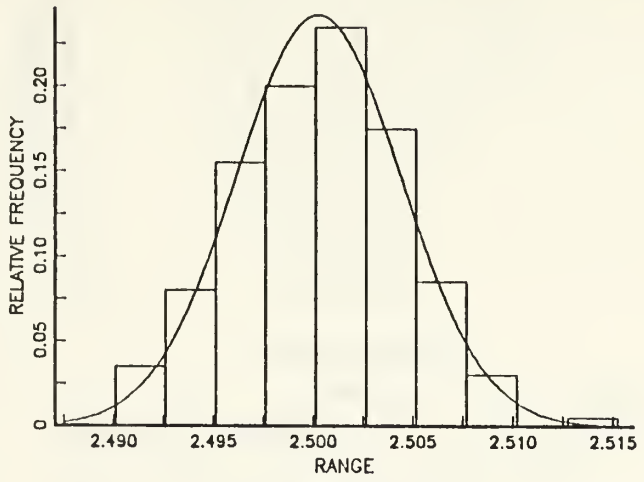


TRACK 3 SENSOR 55

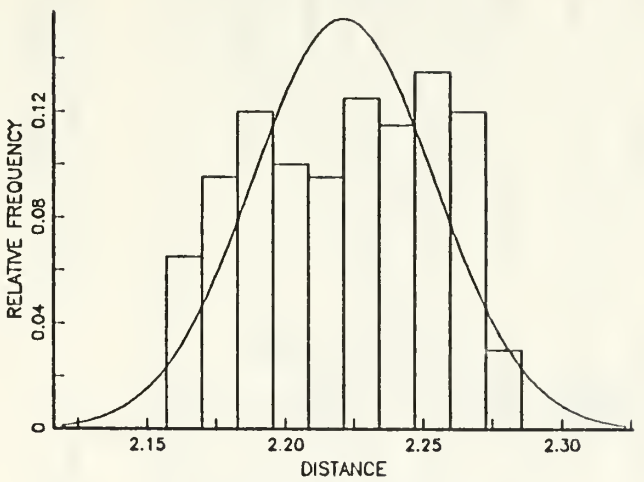


CONSTANT ERROR: MINUS 0.0005

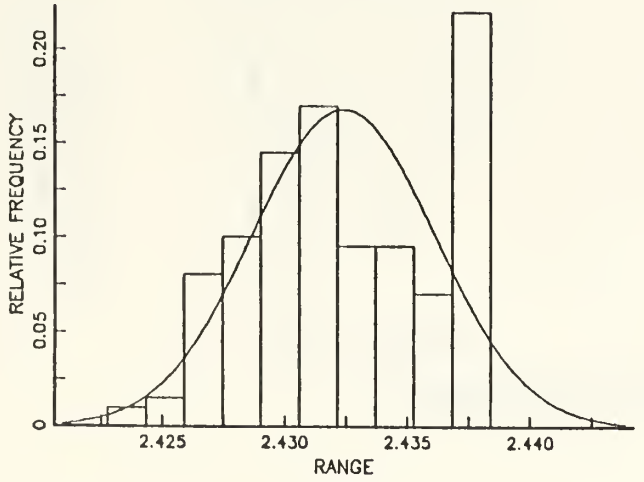
TRACK 4 SENSOR 54



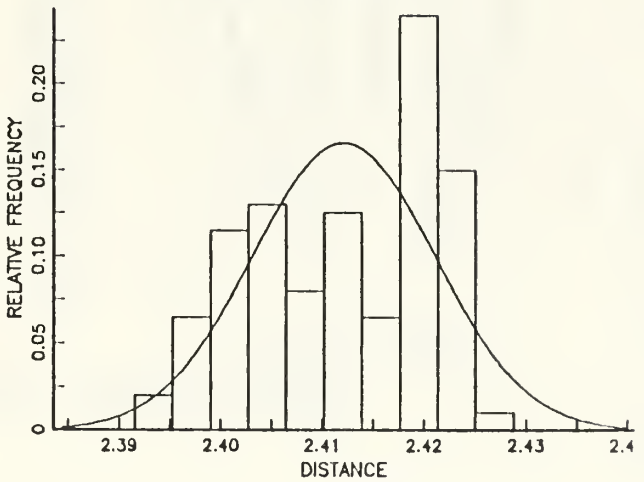
TRACK 4 SENSOR 54



TRACK 4 SENSOR 55

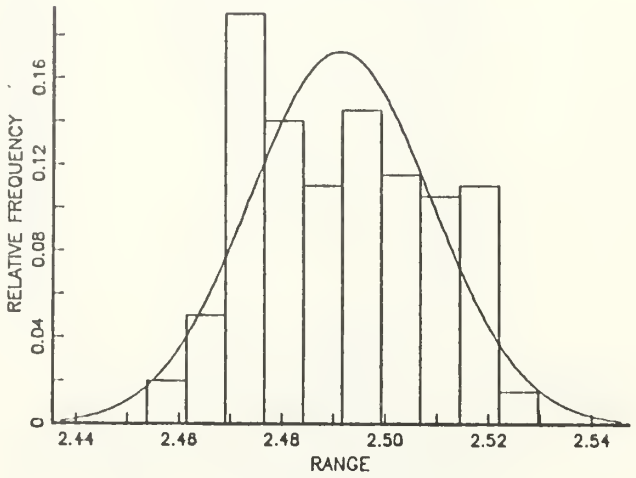


TRACK 4 SENSOR 55

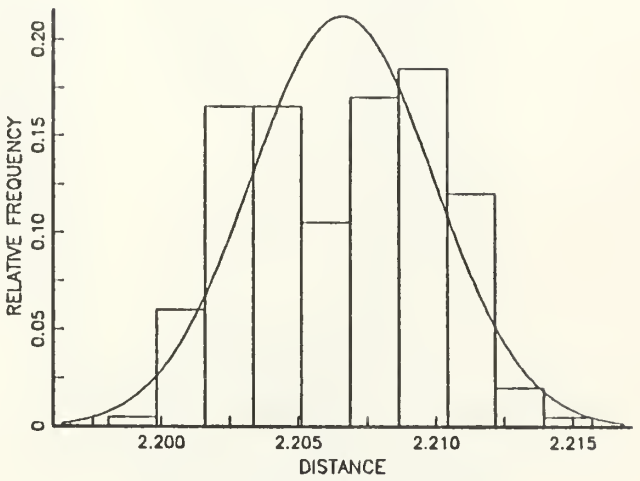


CONSTANT ERROR: MINUS 0.0005

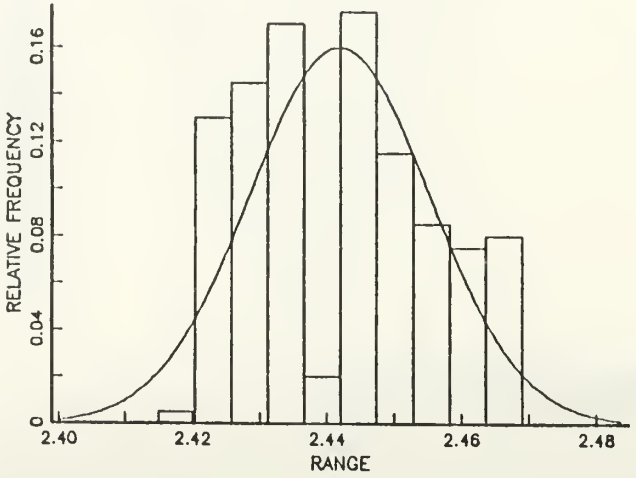
TRACK 5 SENSOR 54



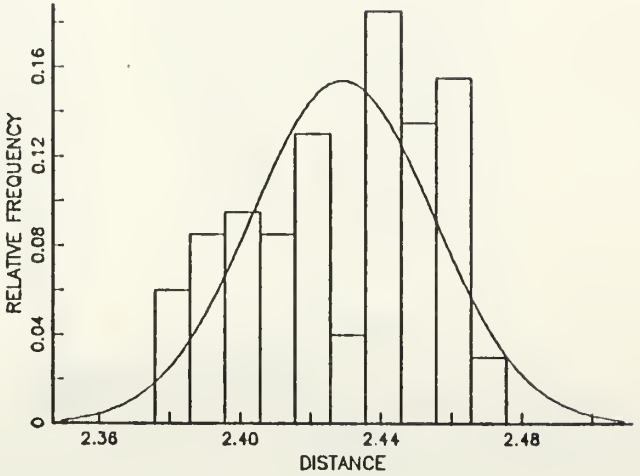
TRACK 5 SENSOR 54



TRACK 5 SENSOR 55



TRACK 5 SENSOR 55



APPENDIX I

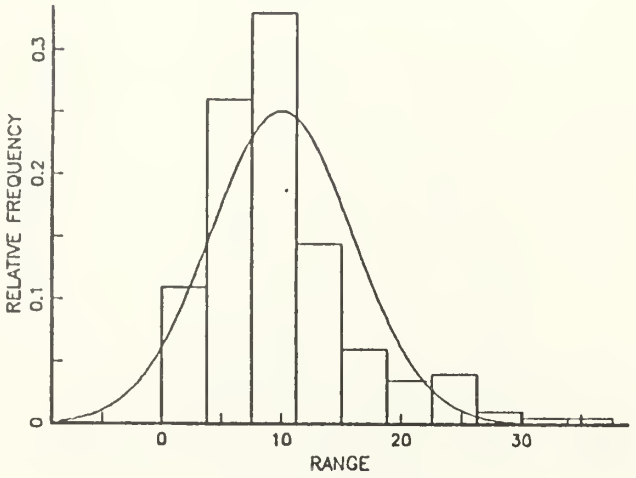
This appendix contains the graphs for the Normal Time Error model (Case IV). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

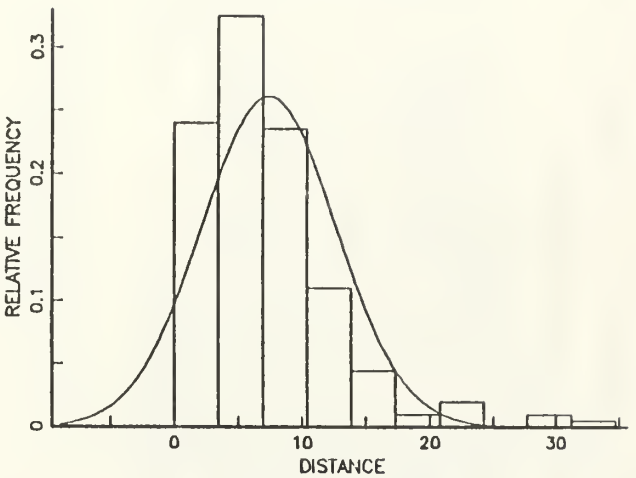
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

NORMAL ERROR: STD DEVIATION 0.00001

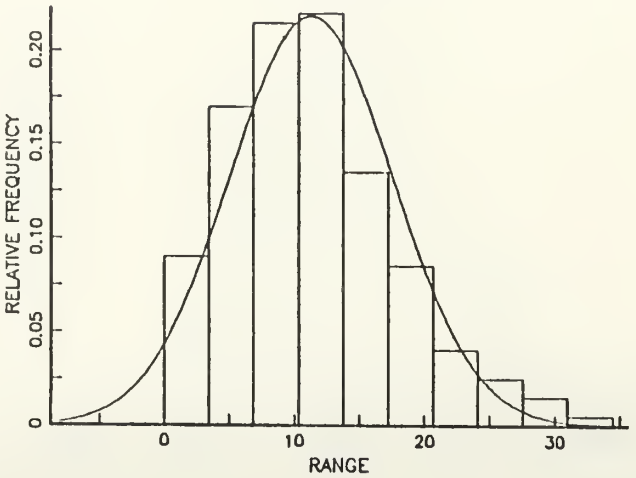
TRACK 1 SENSOR 54



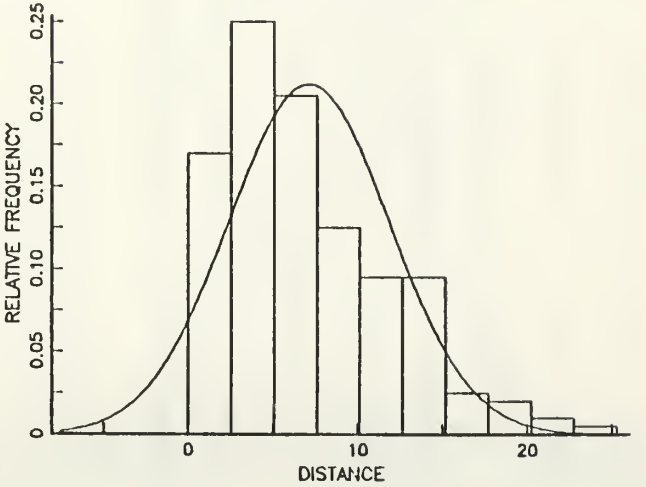
TRACK 1 SENSOR 54



TRACK 1 SENSOR 55

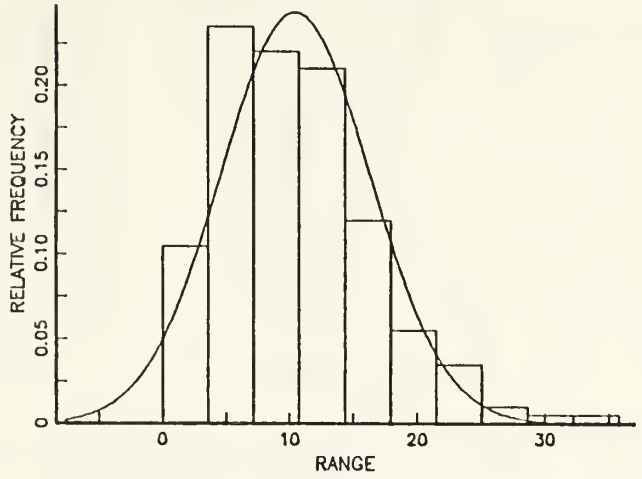


TRACK 1 SENSOR 55

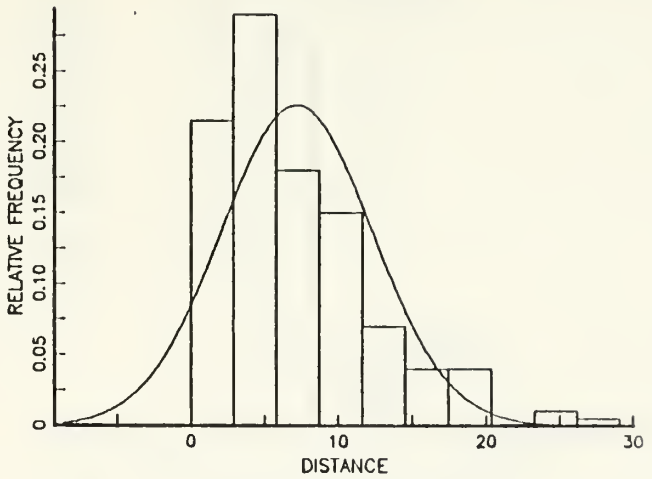


NORMAL ERROR: STD DEVIATION 0.00001

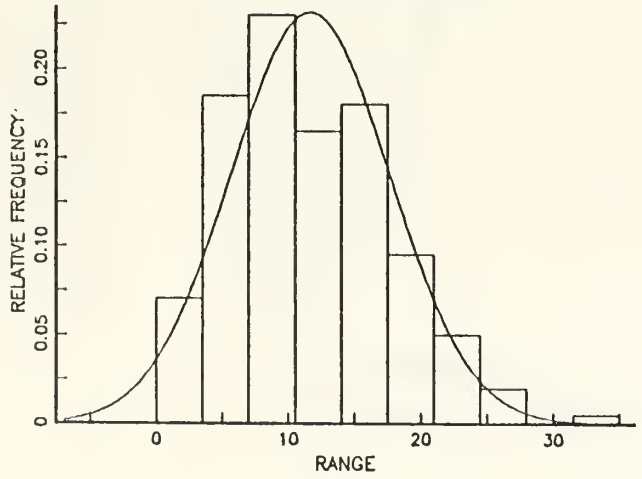
TRACK 2 SENSOR 54



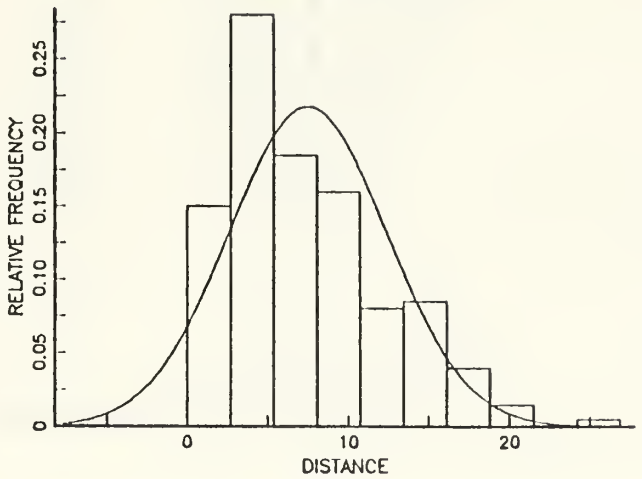
TRACK 2 SENSOR 54



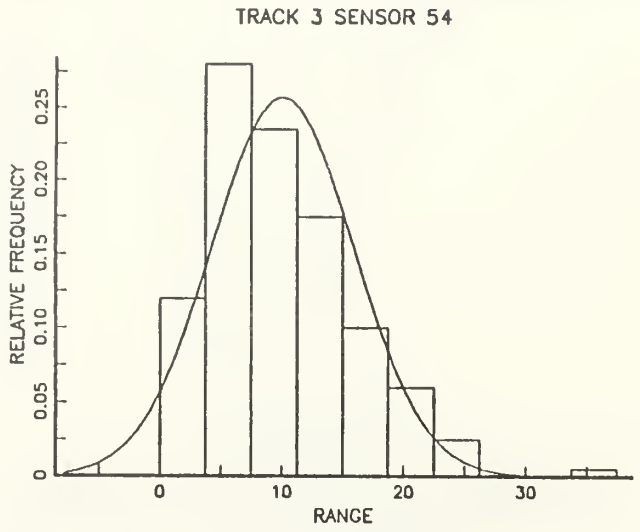
TRACK 2 SENSOR 55

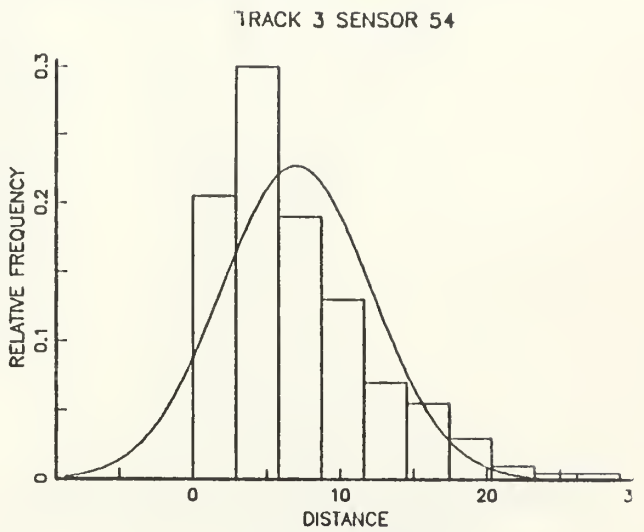


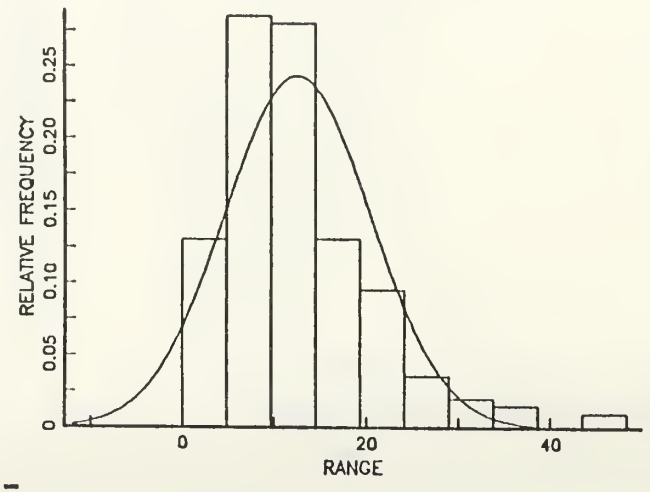
TRACK 2 SENSOR 55

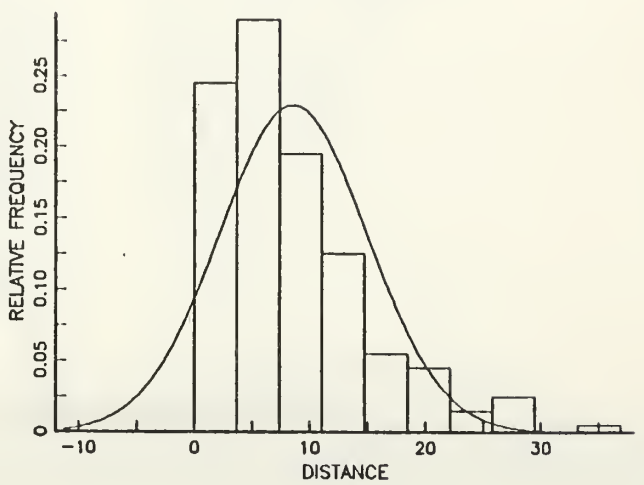


NORMAL ERROR: STD DEVIATION 0.00001

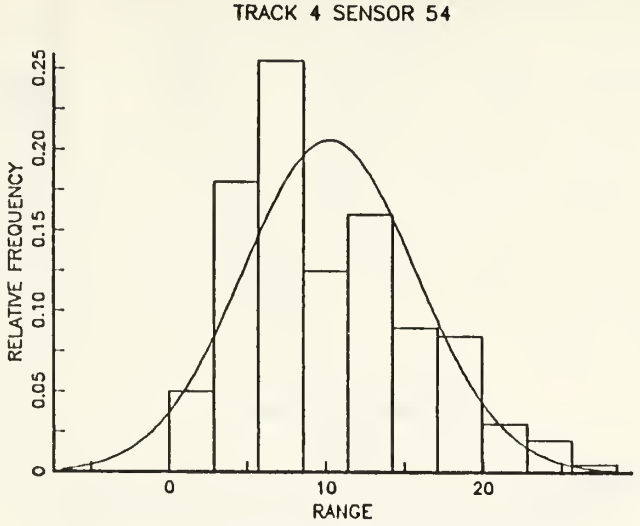


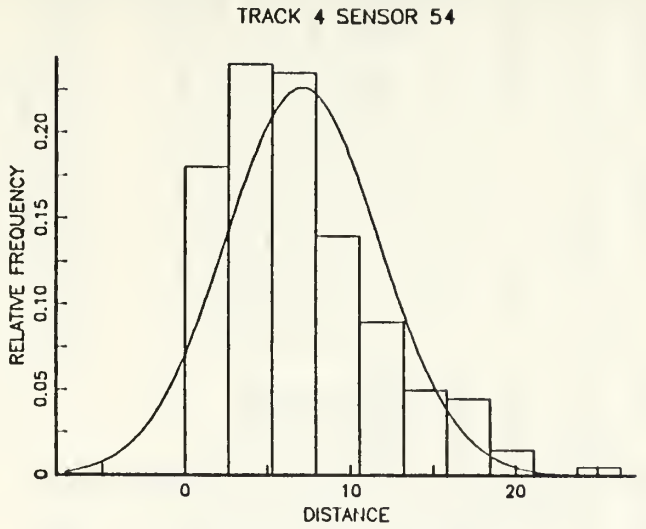


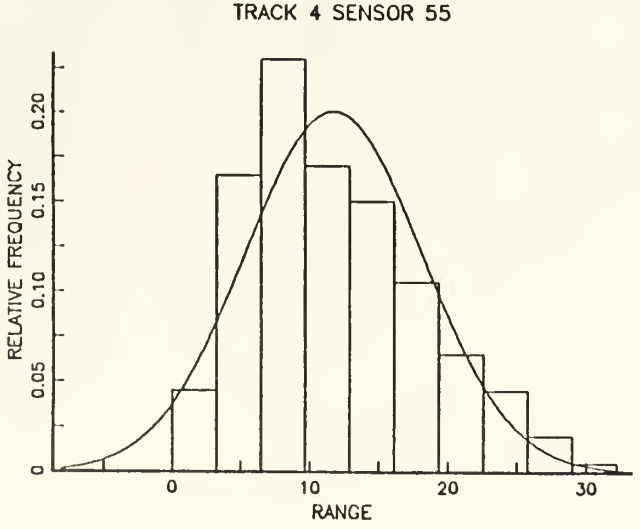


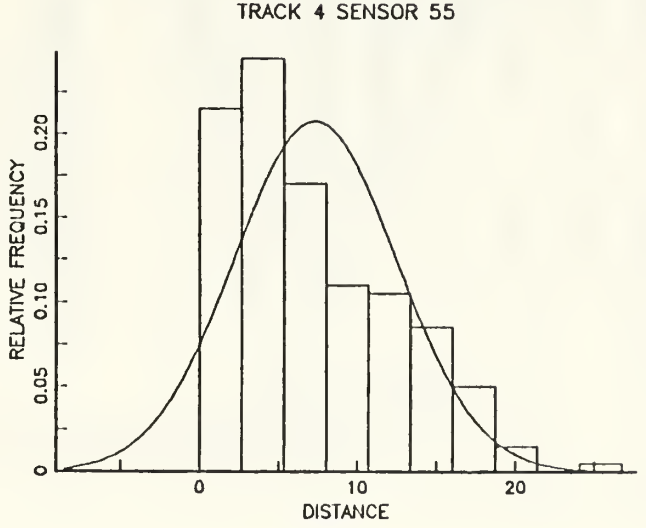


NORMAL ERROR: STD DEVIATION 0.00001



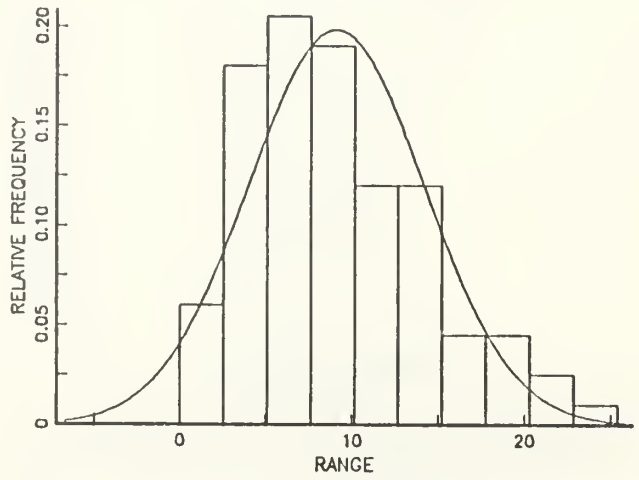




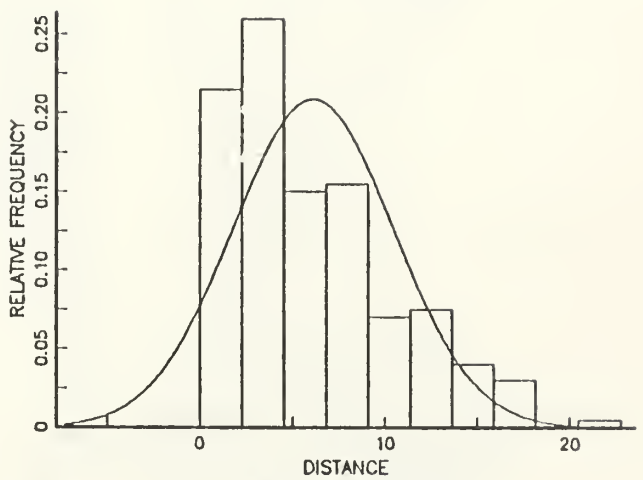


NORMAL ERROR: STD DEVIATION 0.00001

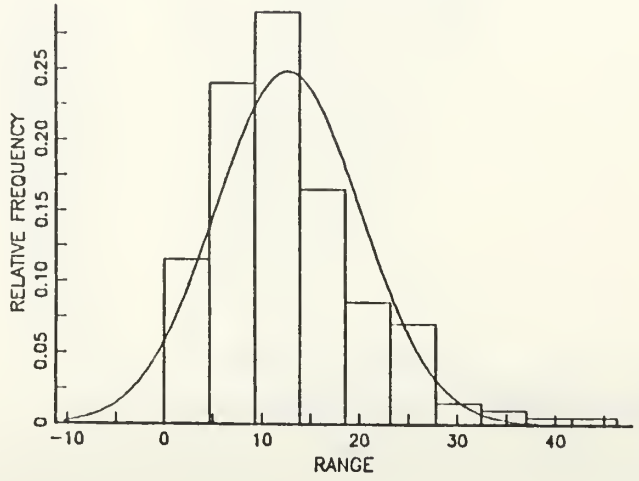
TRACK 5 SENSOR 54



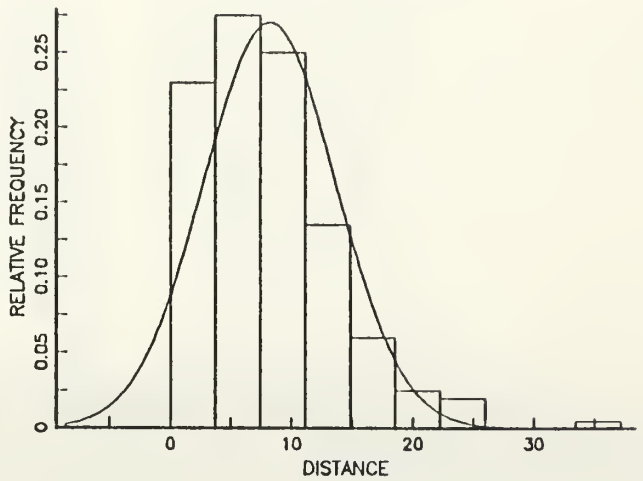
TRACK 5 SENSOR 54



TRACK 5 SENSOR 55



TRACK 5 SENSOR 55



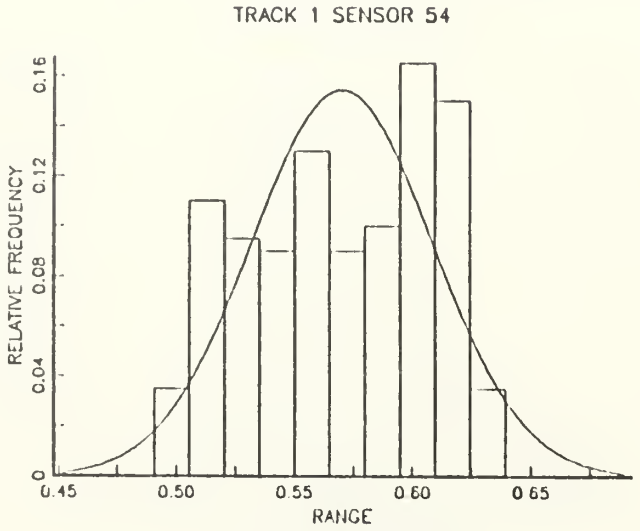
APPENDIX J

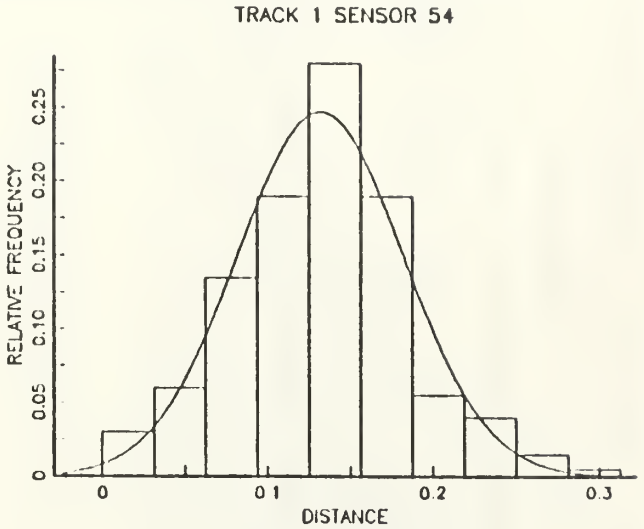
This appendix contains the graphs for the Normal Time Error model (Case V). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

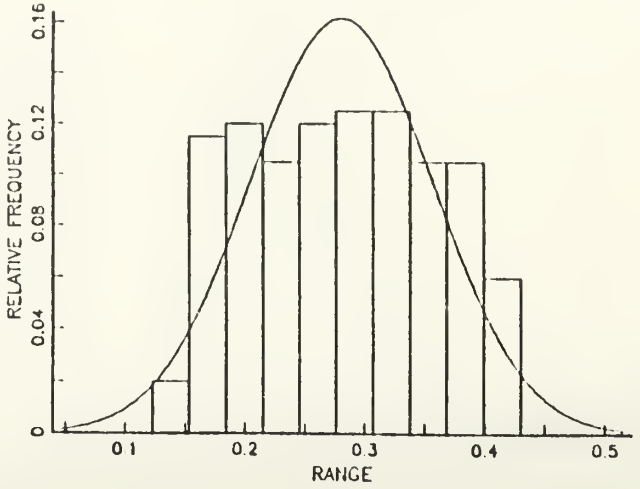
"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

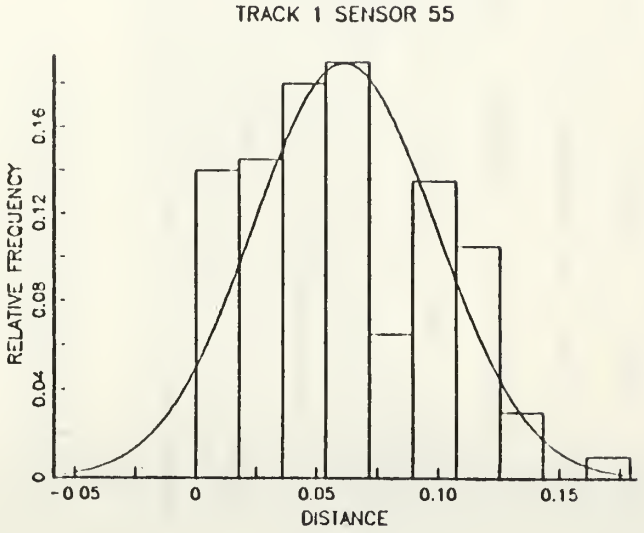
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

Equal Time Error at Each Hydrophone: St Dev 0.00001

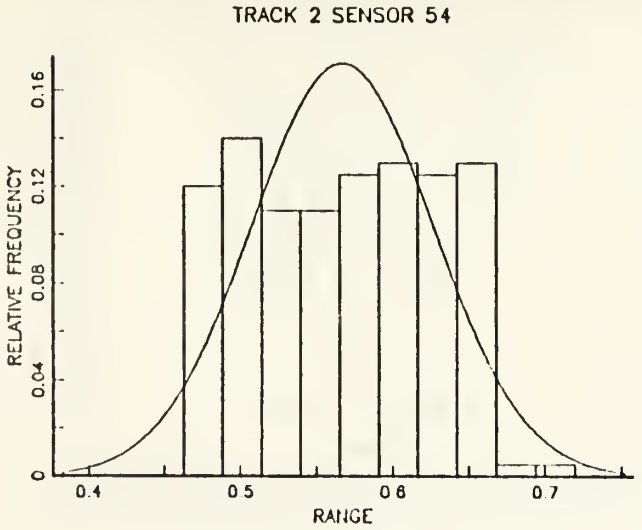


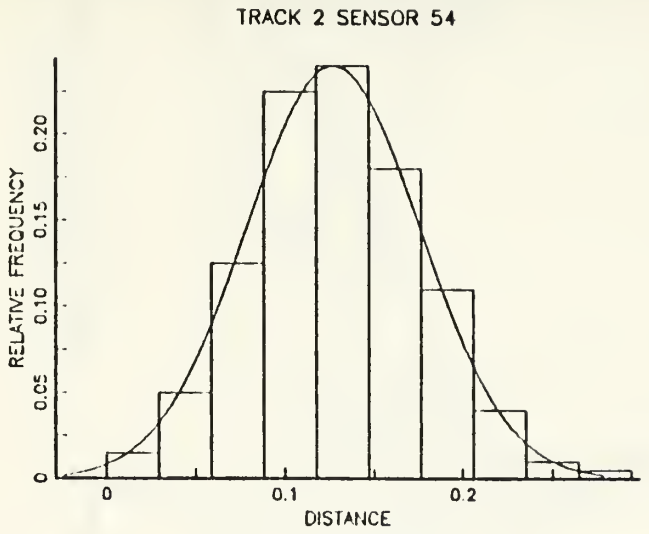


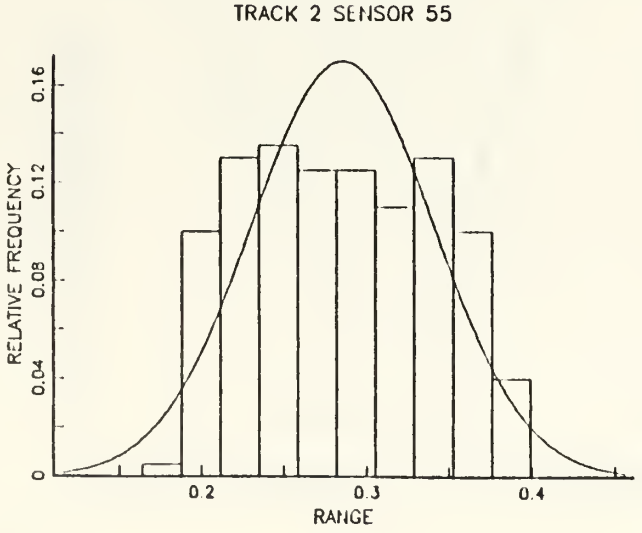


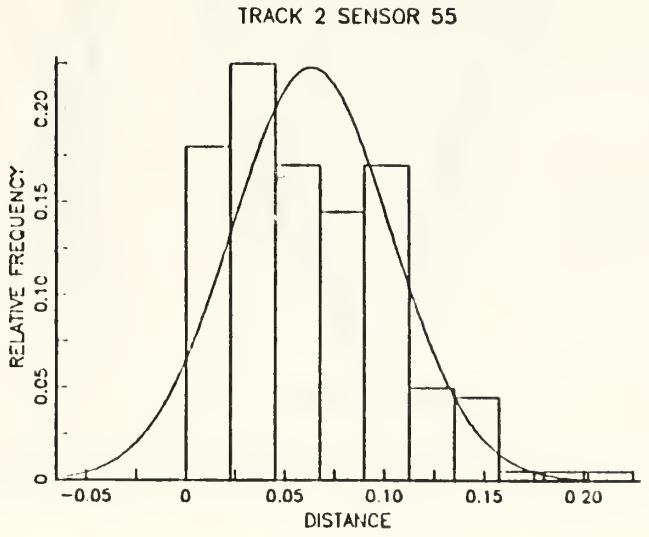


Equal Time Error at Each Hydrophone: St Dev 0.00001

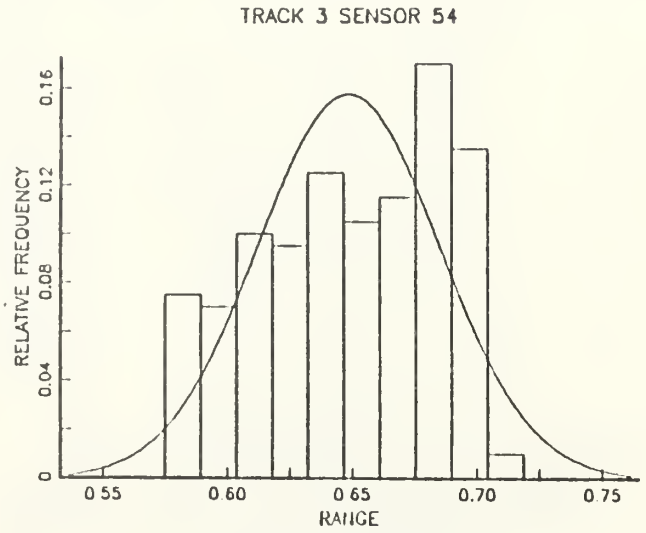


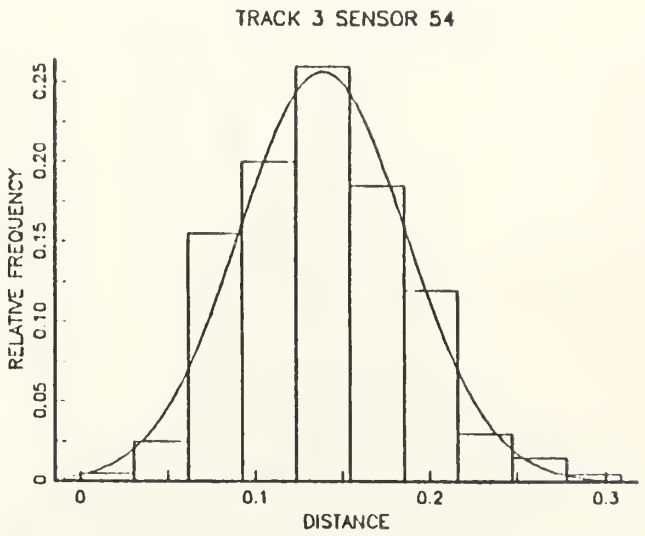


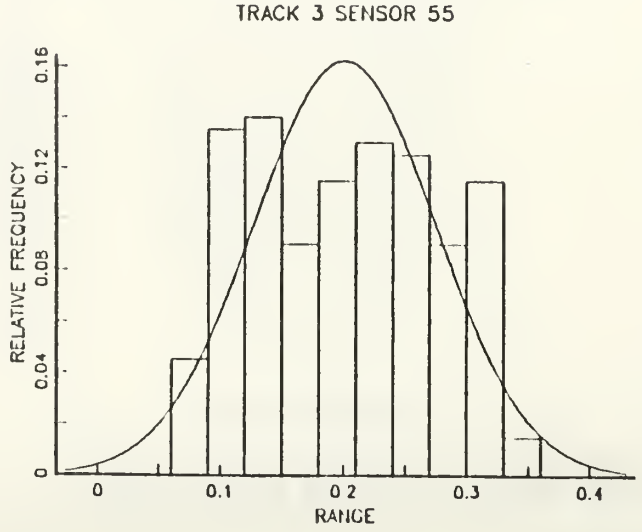


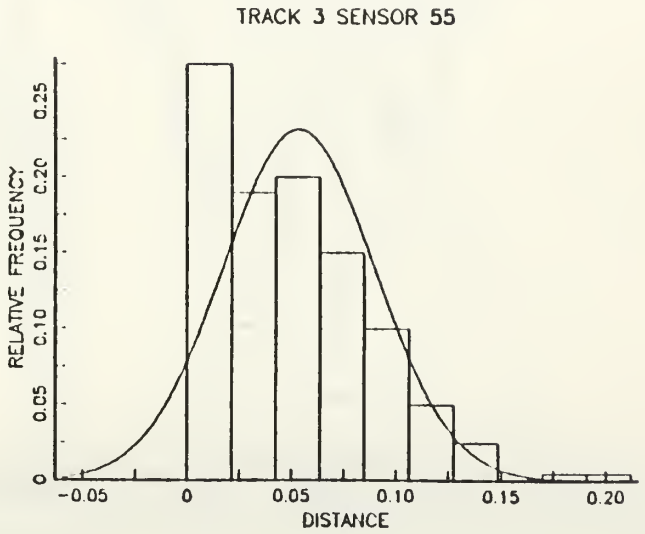


Equal Time Error at Each Hydrophone: St Dev 0.00001

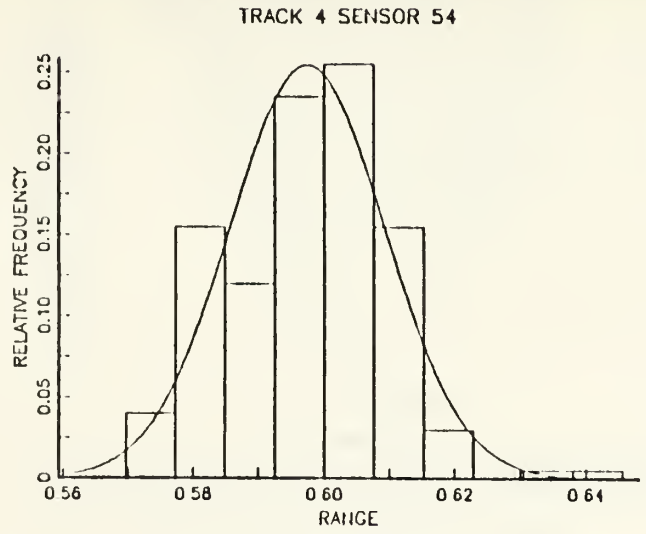


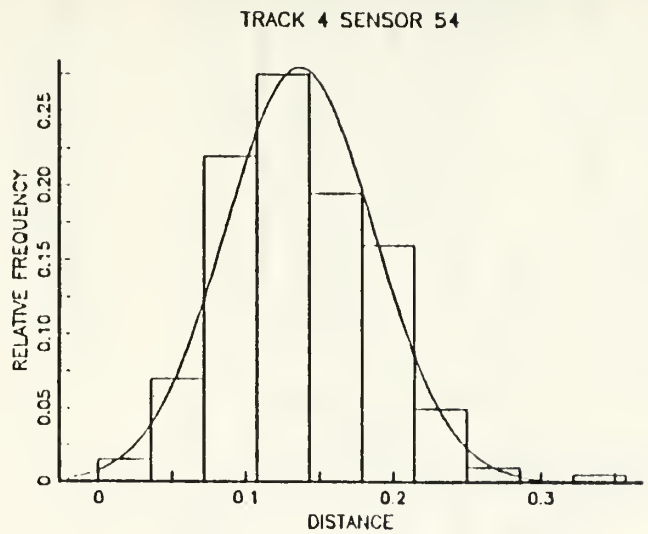


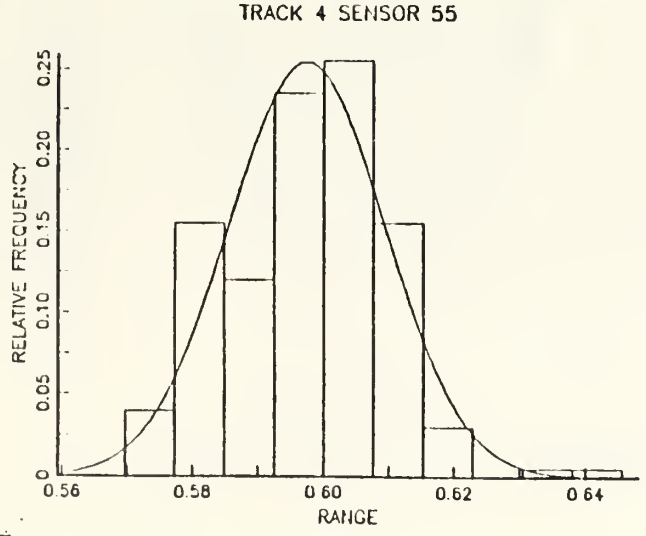


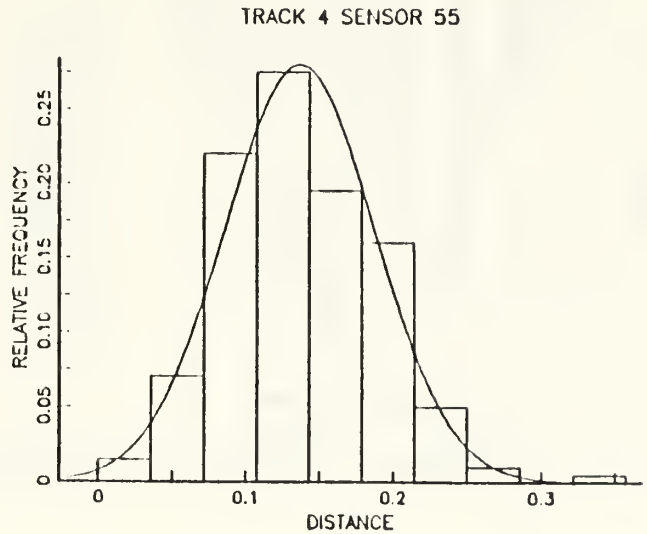


Equal Time Error at Each Hydrophone: St Dev 0.00001

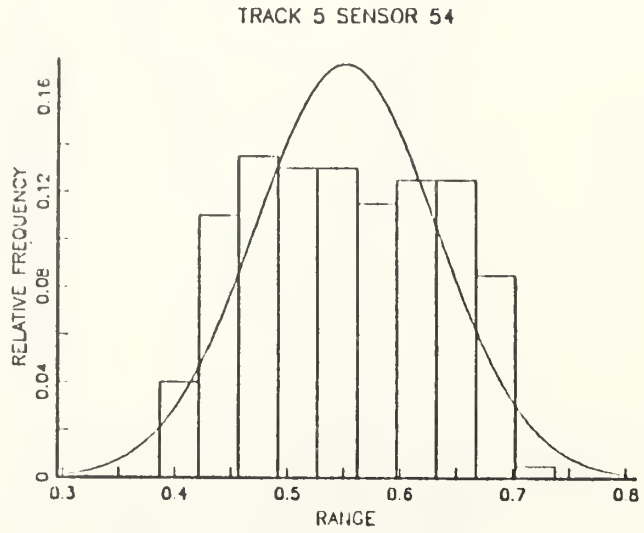


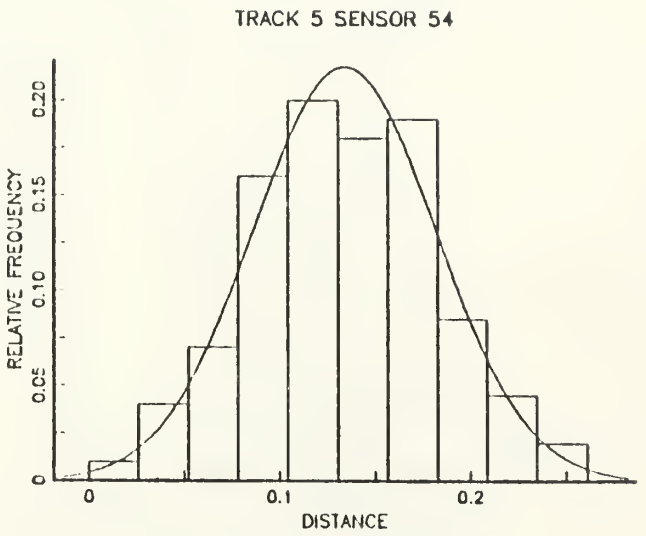


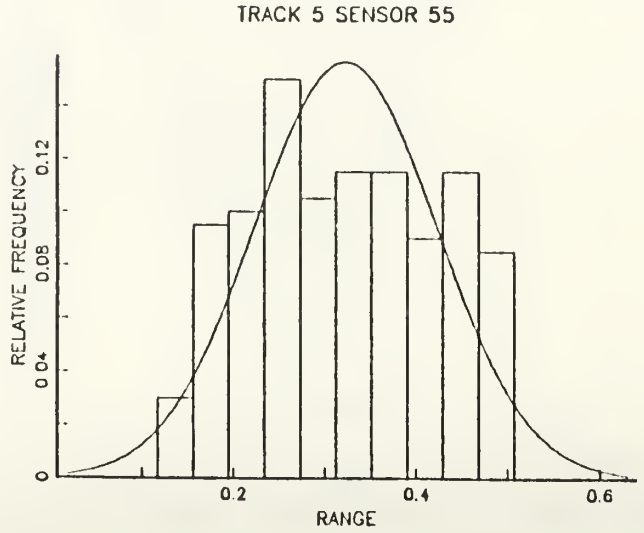


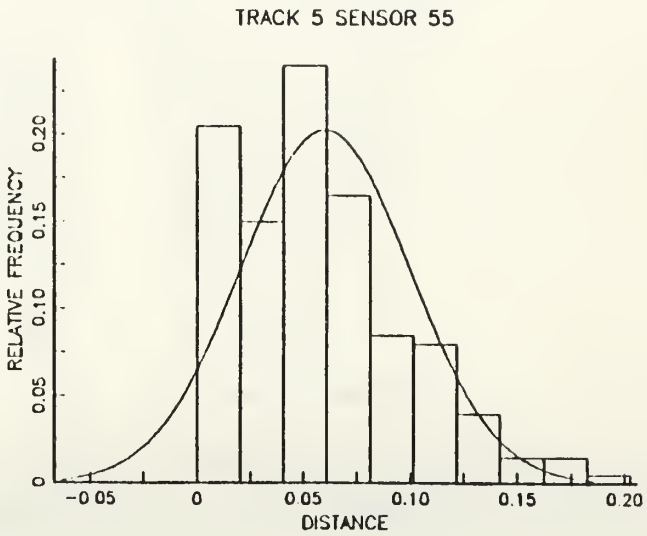


Equal Time Error at Each Hydrophone: St Dev 0.00001









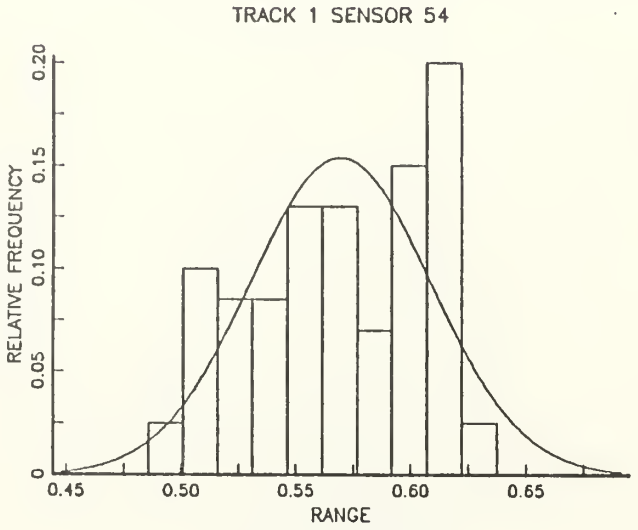
APPENDIX K

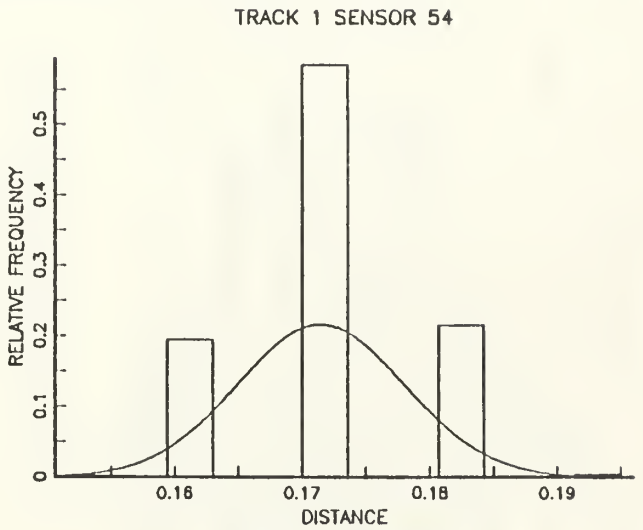
This appendix contains the graphs for the Constant Velocity Error model (Case VI). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

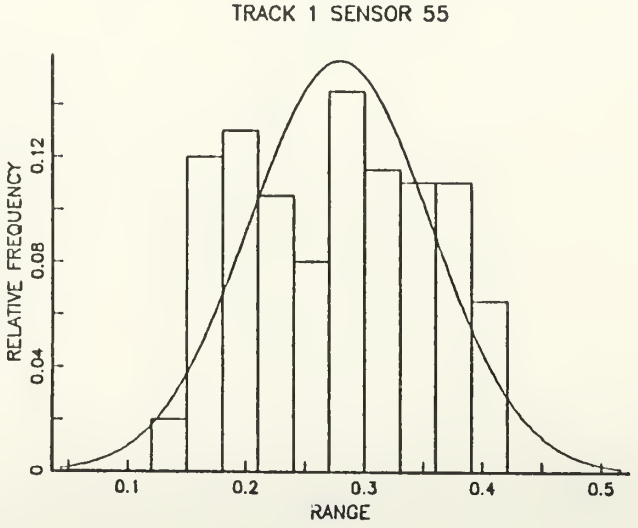
"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

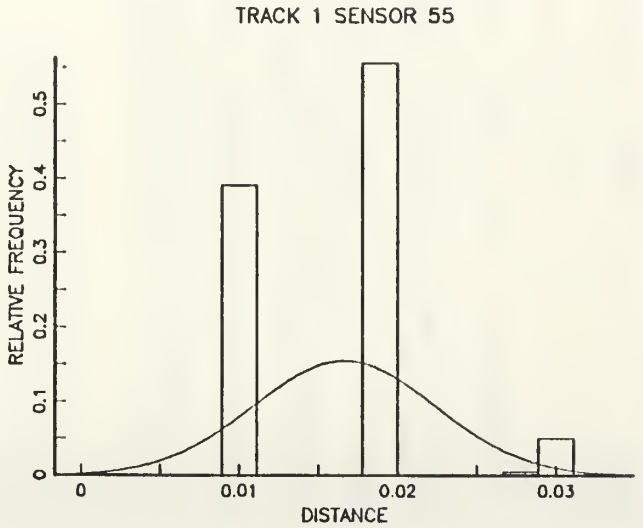
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

CONSTANT VELOCITY ERRORS: PLUS 0.005



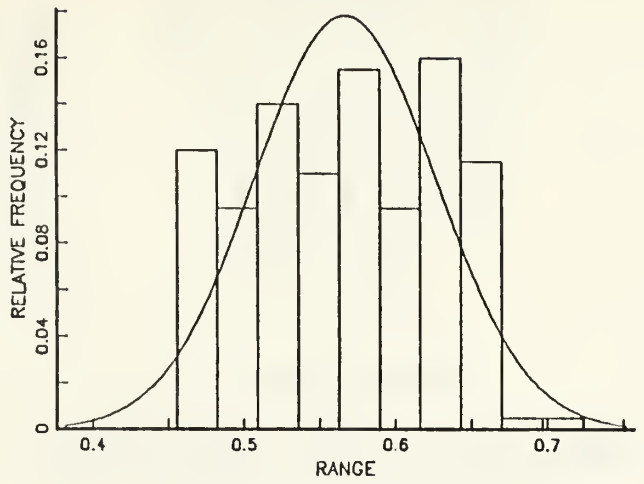




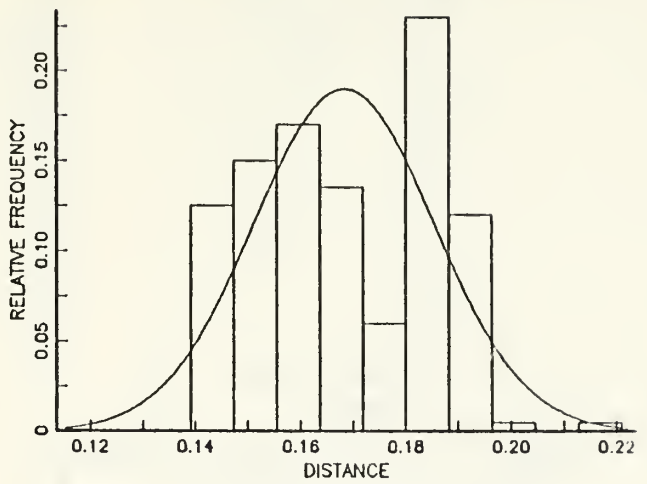


CONSTANT VELOCITY ERRORS: PLUS 0.005

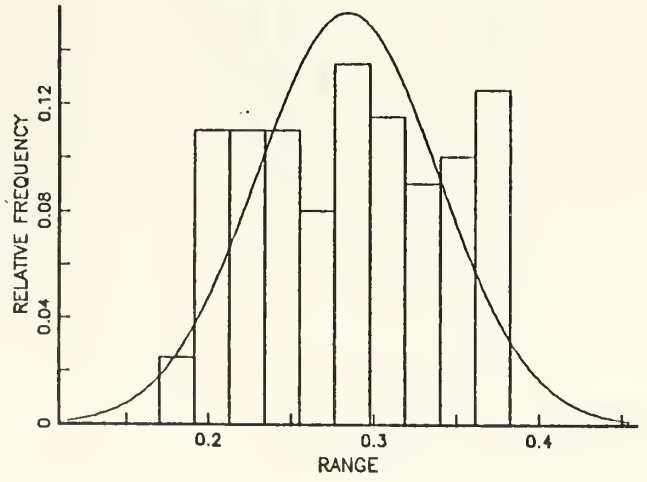
TRACK 2 SENSOR 54



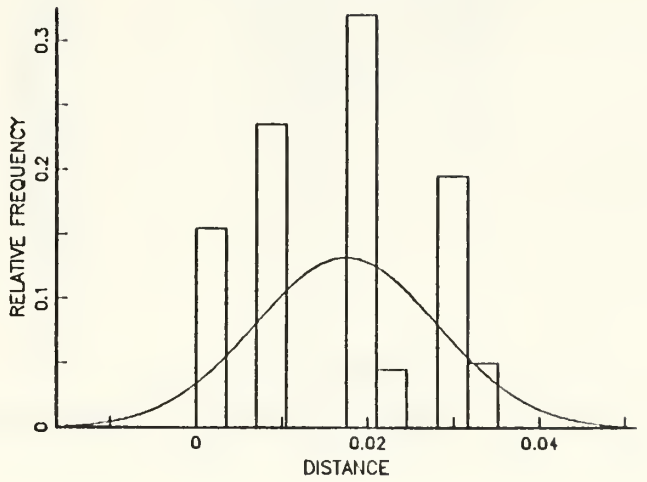
TRACK 2 SENSOR 54



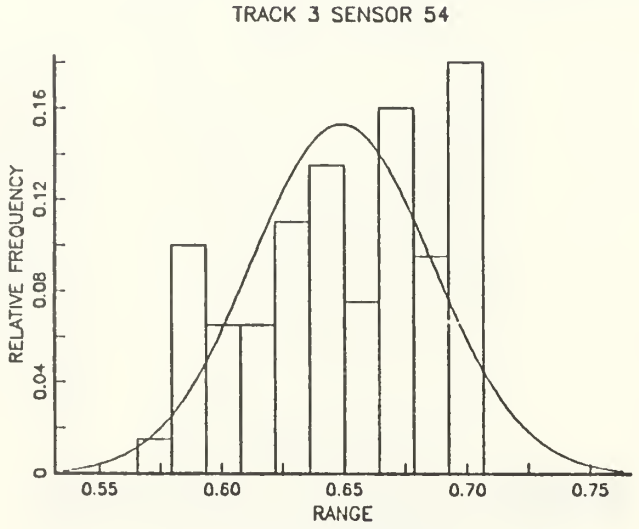
TRACK 2 SENSOR 55

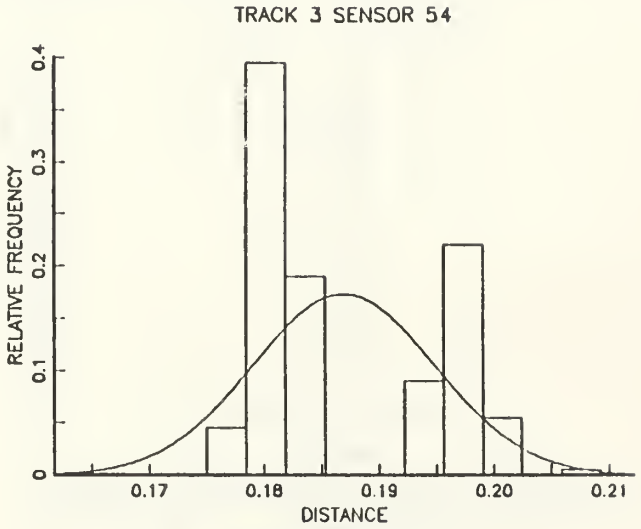


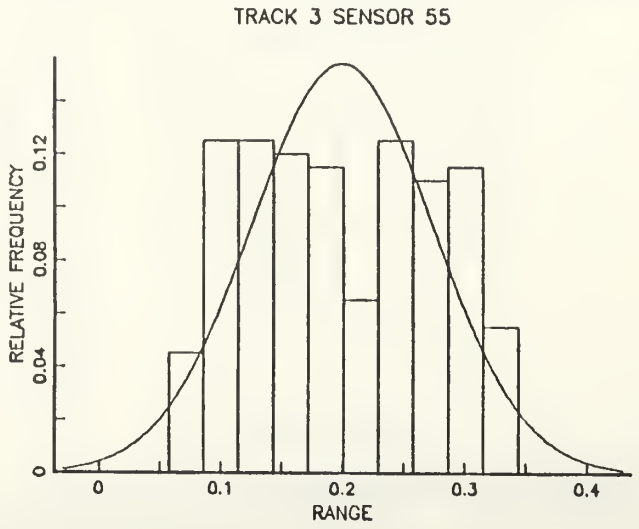
TRACK 2 SENSOR 55

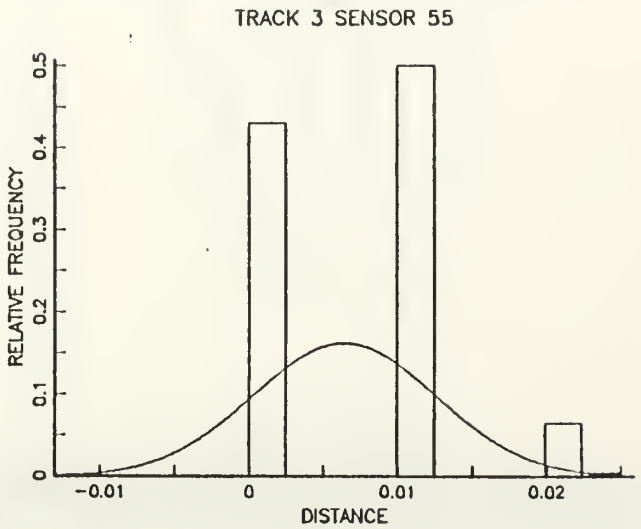


CONSTANT VELOCITY ERRORS: PLUS 0.005

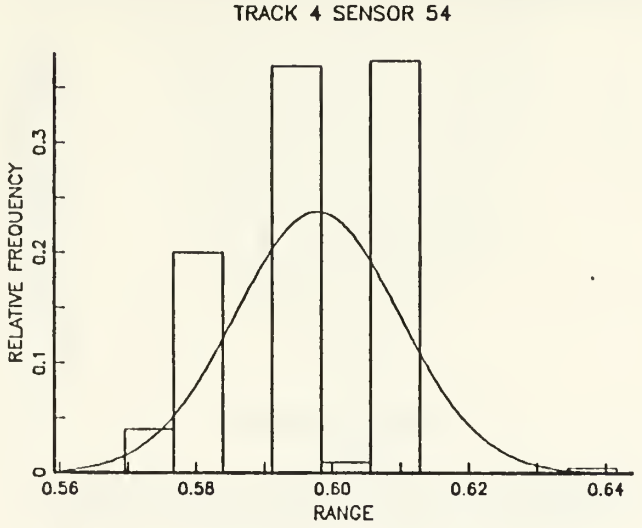


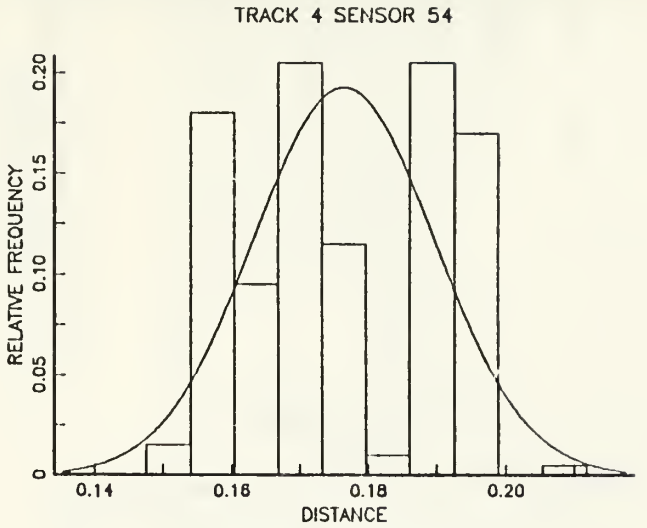


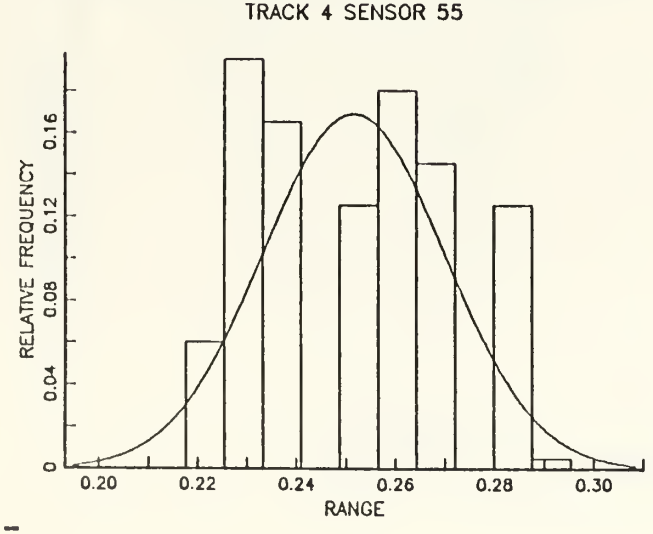


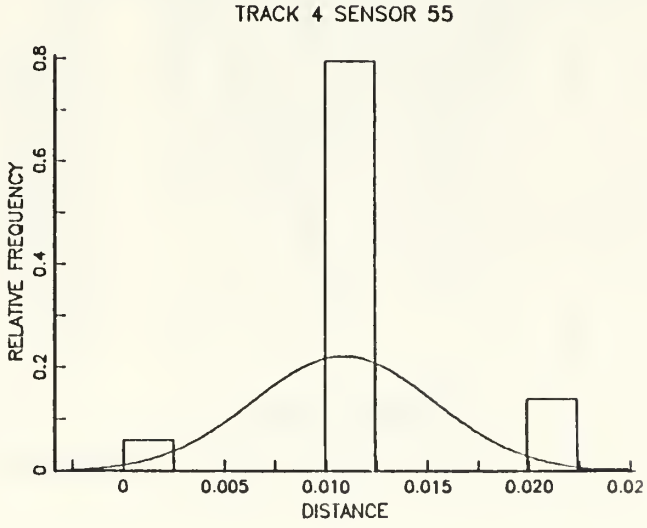


CONSTANT VELOCITY ERRORS: PLUS 0.005

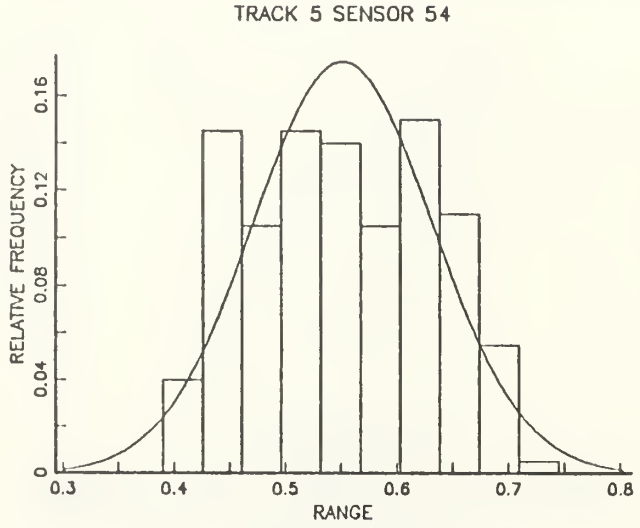


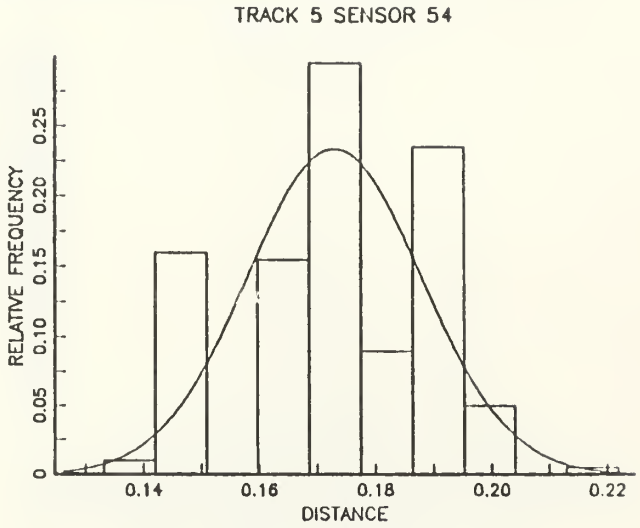


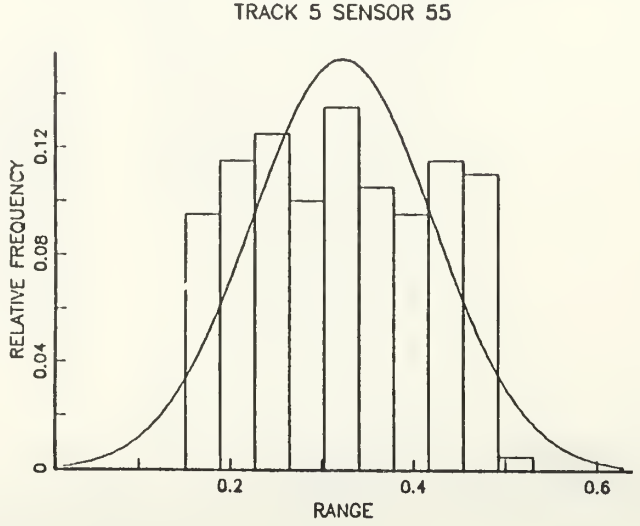


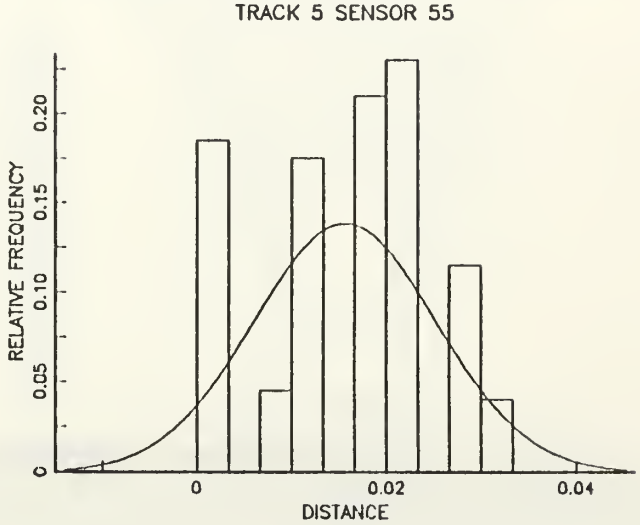


CONSTANT VELOCITY ERRORS: PLUS 0.005









APPENDIX L

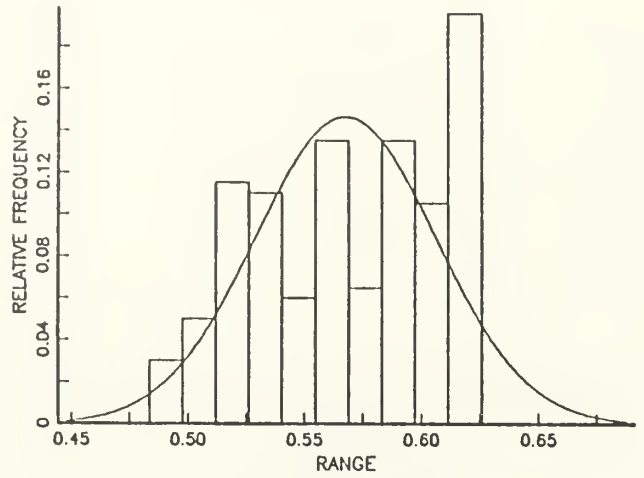
This appendix contains the graphs for the Constant Velocity Error model (Case VII). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

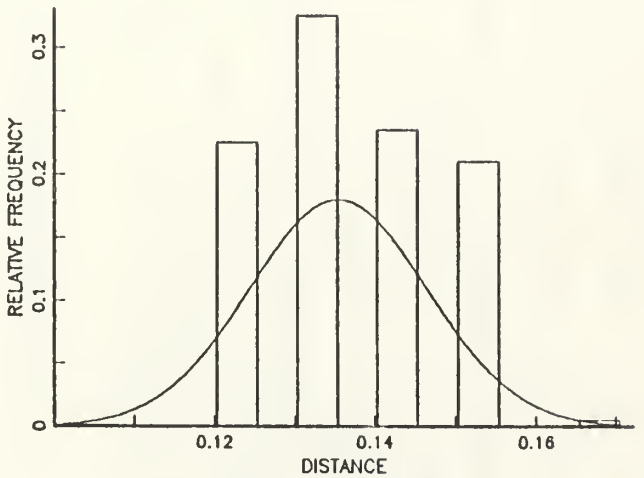
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

CONSTANT VELOCITY ERRORS: PLUS 0.05

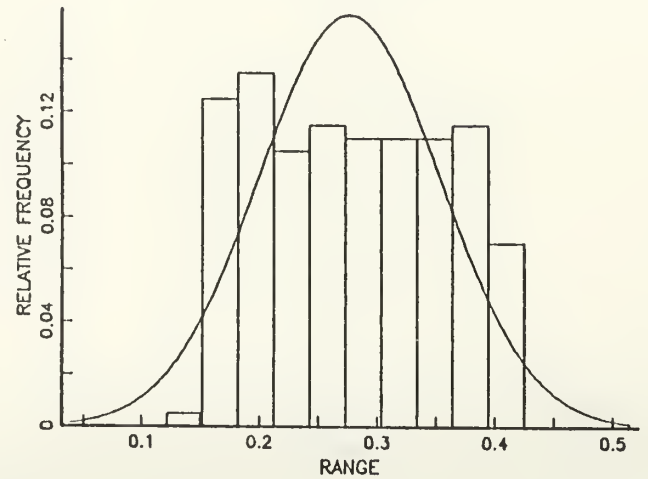
TRACK 1 SENSOR 54



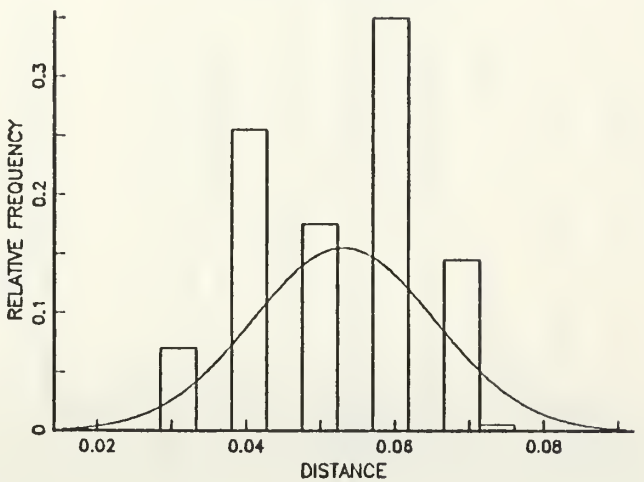
TRACK 1 SENSOR 54



TRACK 1 SENSOR 55

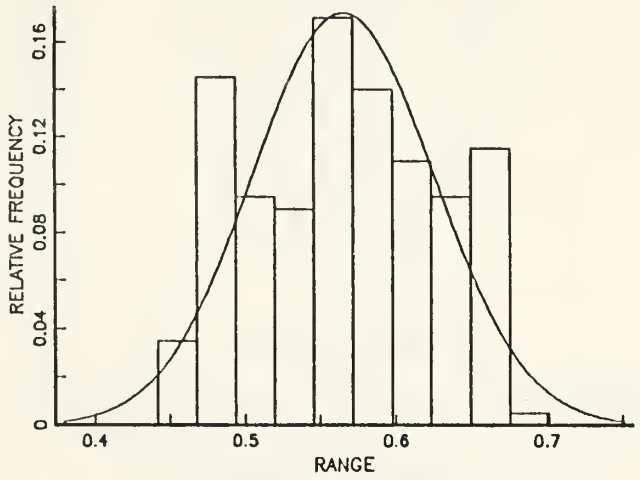


TRACK 1 SENSOR 55

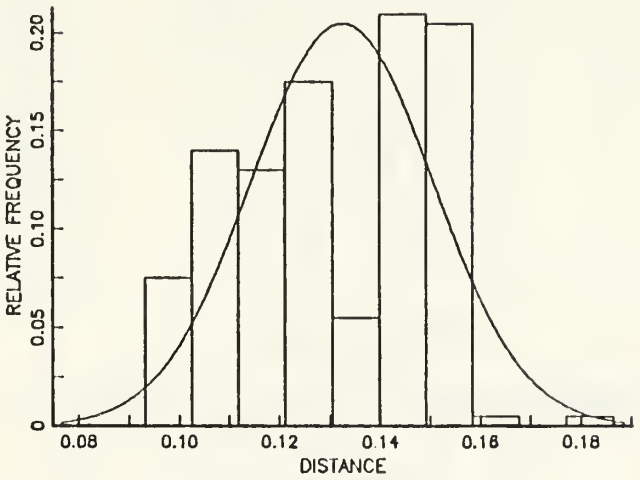


CONSTANT VELOCITY ERRORS: PLUS 0.05

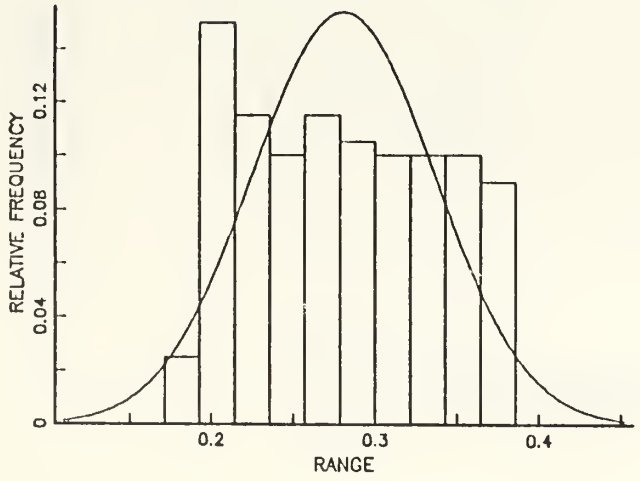
TRACK 2 SENSOR 54



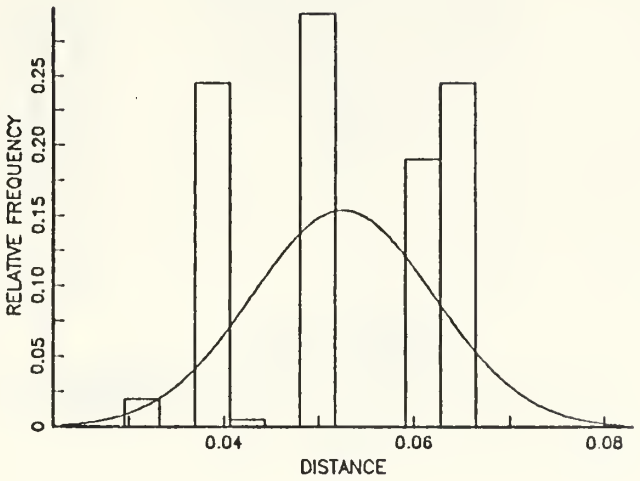
TRACK 2 SENSOR 54



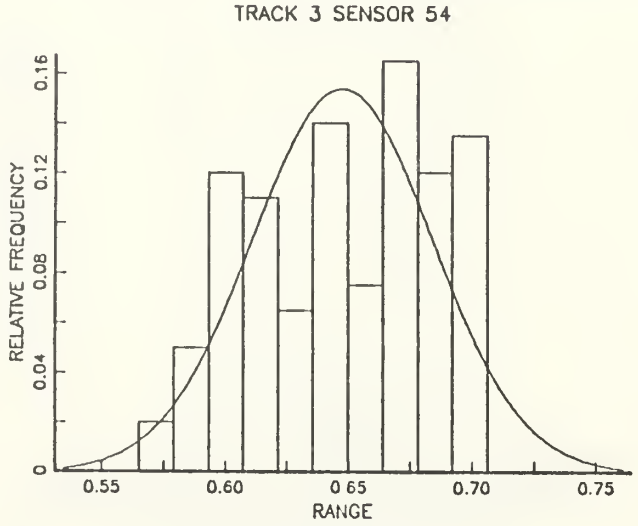
TRACK 2 SENSOR 55

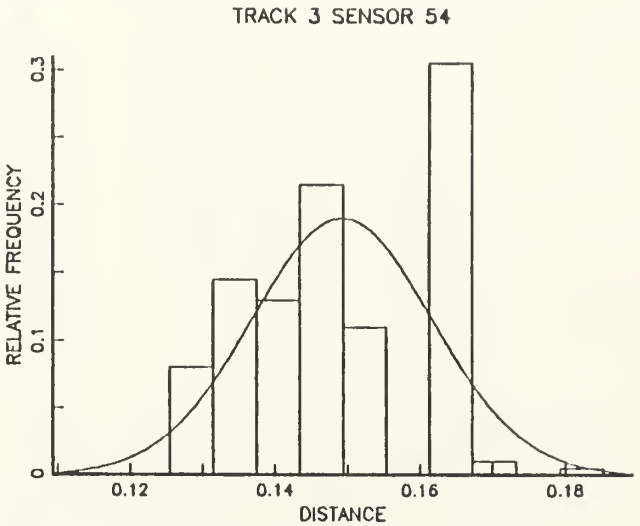


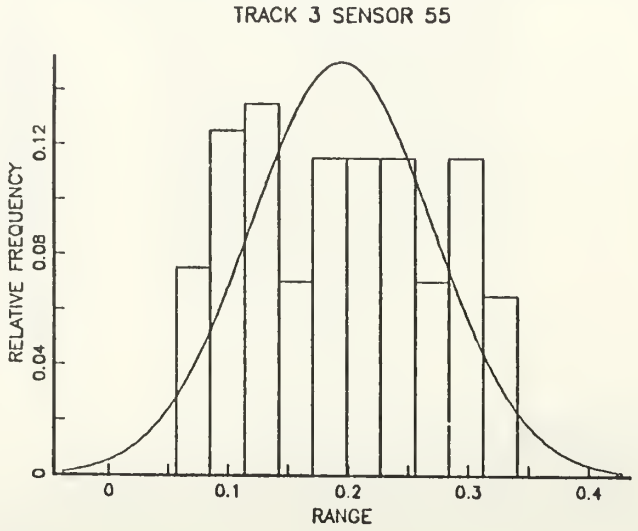
TRACK 2 SENSOR 55

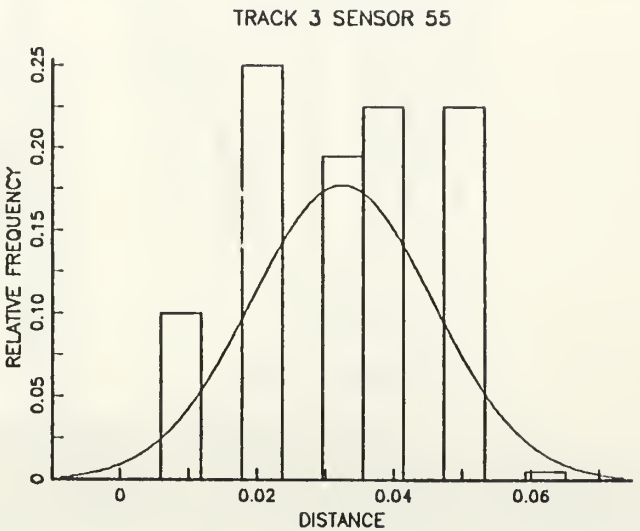


CONSTANT VELOCITY ERRORS: PLUS 0.05

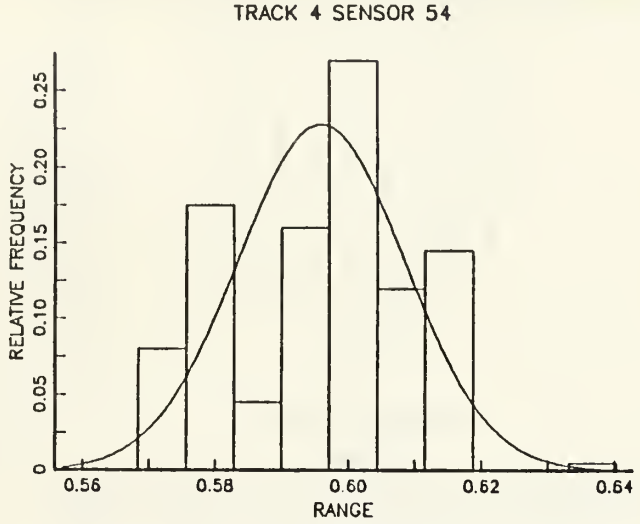


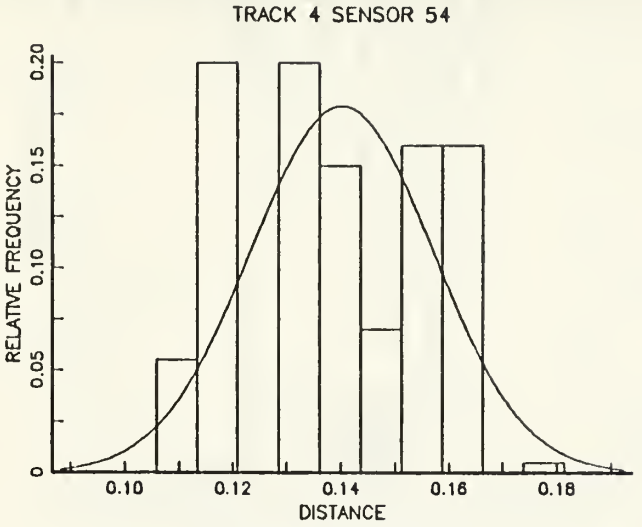


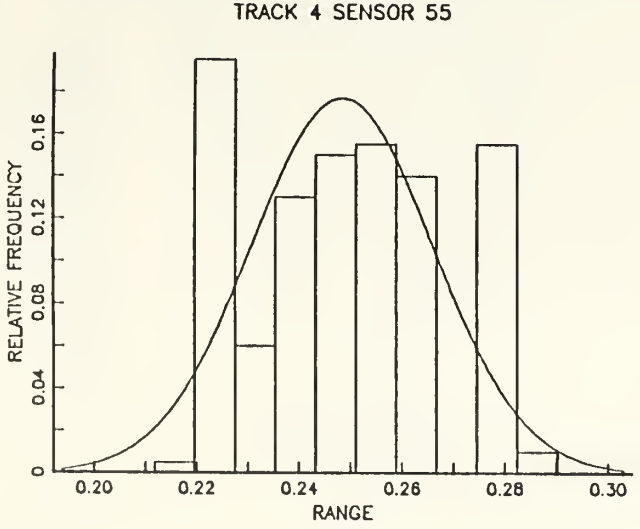


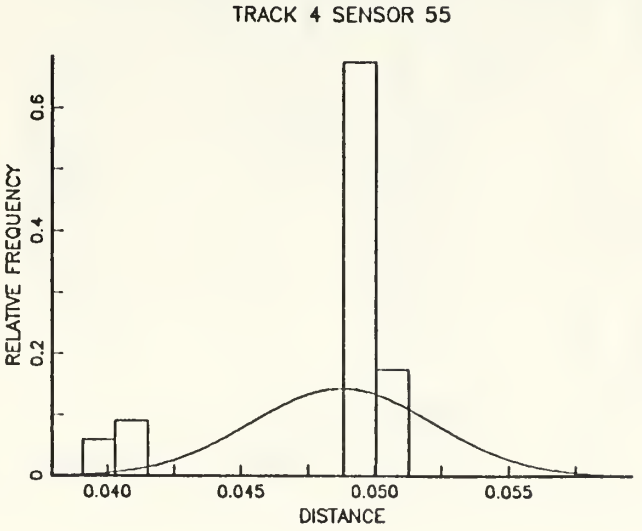


CONSTANT VELOCITY ERRORS: PLUS 0.05

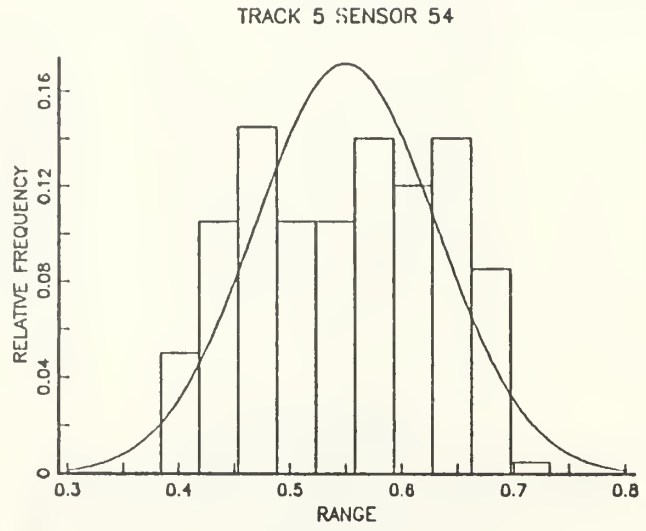


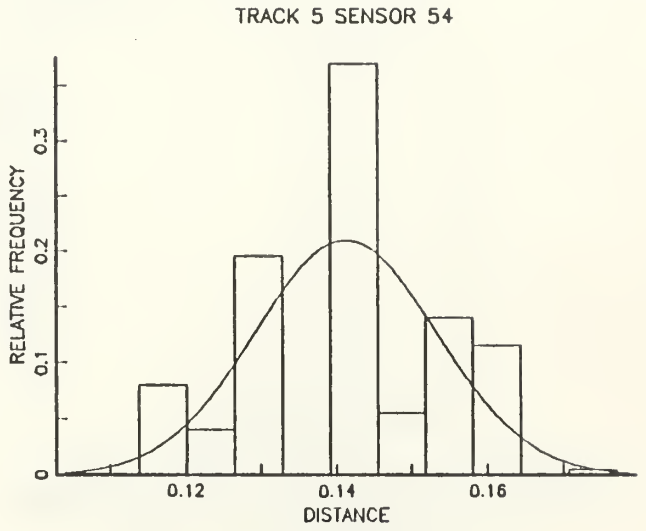


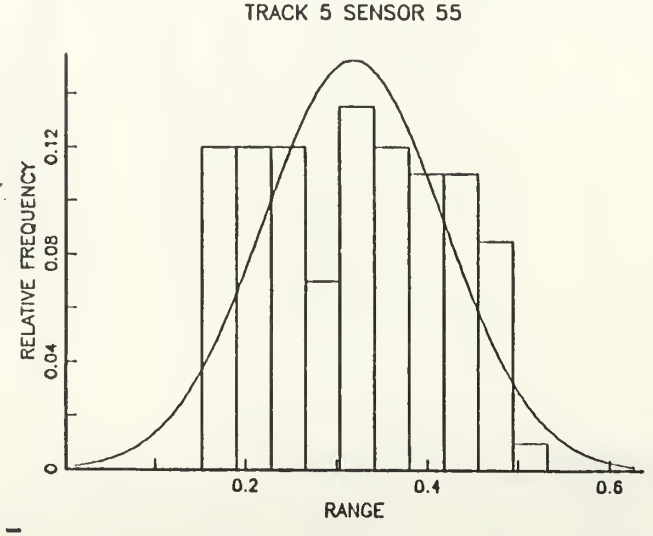


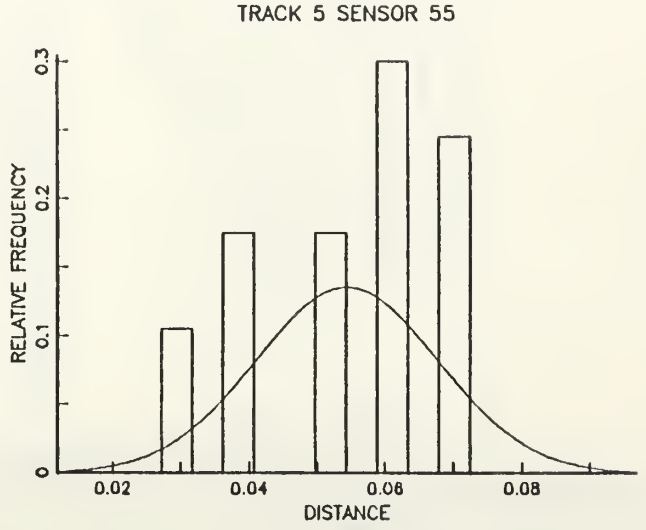


CONSTANT VELOCITY ERRORS: PLUS 0.05









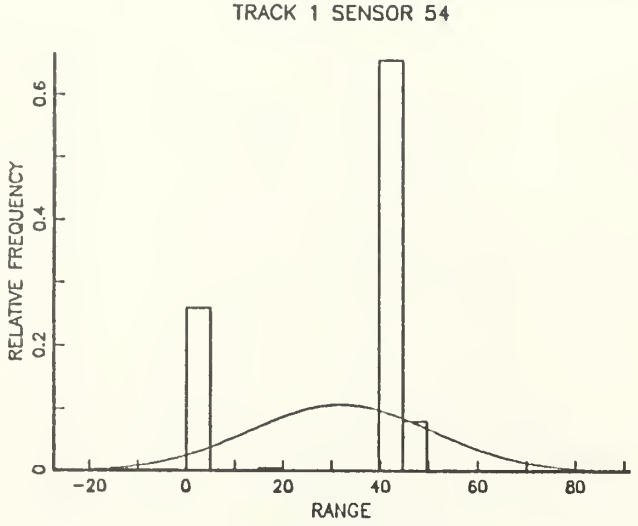
APPENDIX M

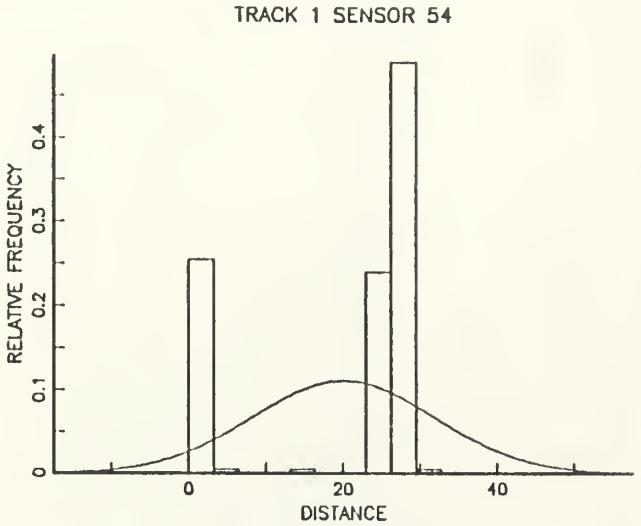
This appendix contains the graphs for the Random Velocity Error model (Case VIII). Four graphs are presented for each track, two for sensor 54 and two for sensor 55.

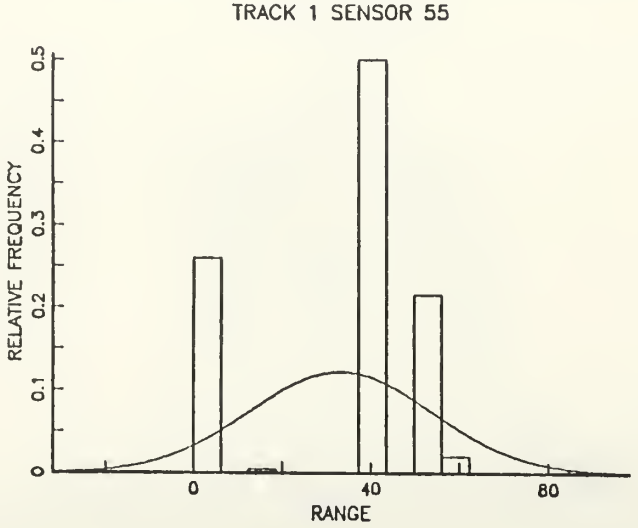
"Range" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a three dimensional distance.

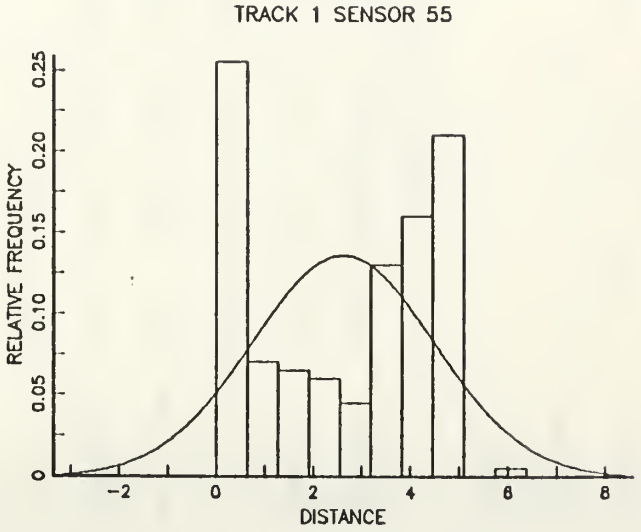
"Distance" in these graphs refers to the miss distance between the raytraced position and the actual position, measured as a two dimensional distance (i.e., a plan view).

NORMAL VELOCITY ERRORS: ST DEV 0.0001



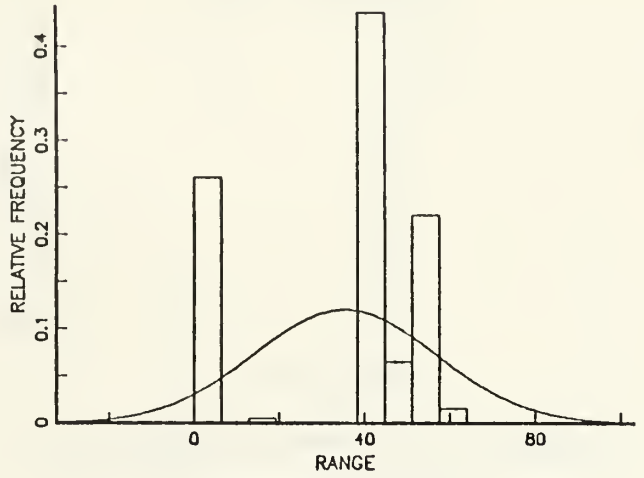




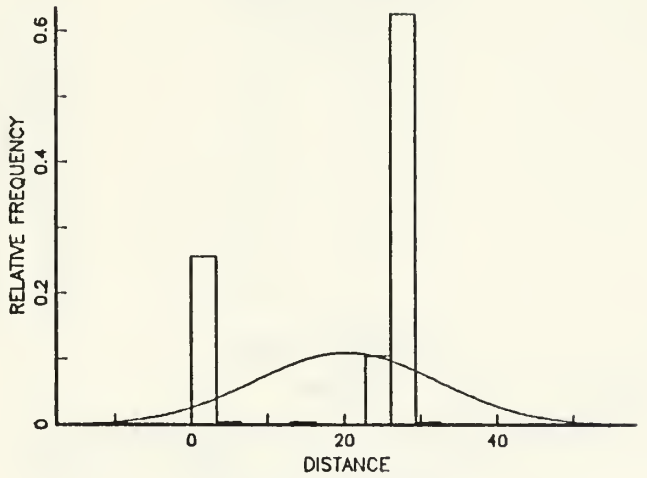


NORMAL VELOCITY ERRORS: ST DEV 0.0001

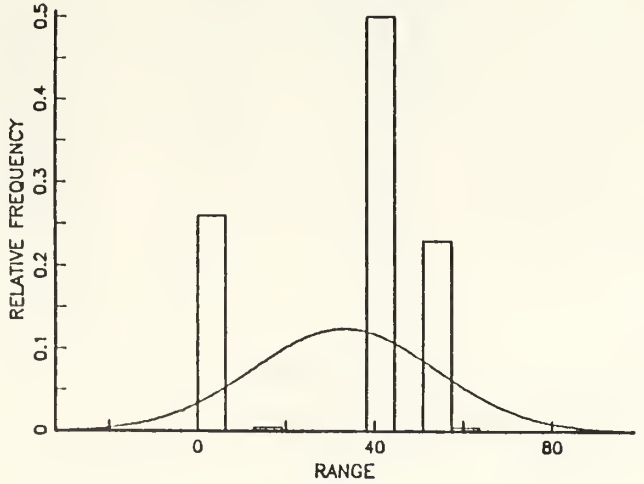
TRACK 2 SENSOR 54



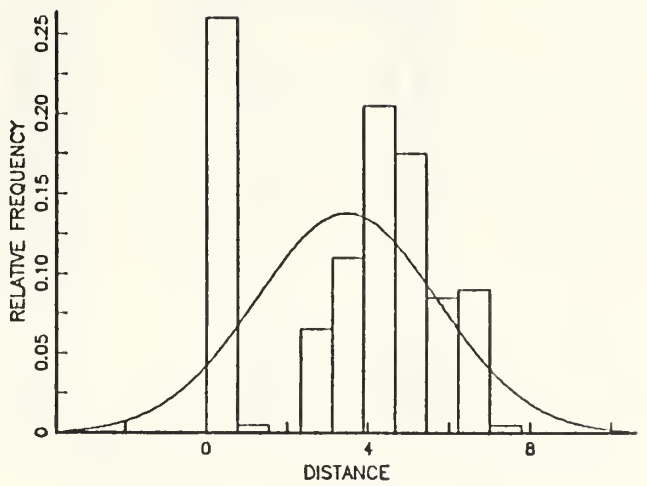
TRACK 2 SENSOR 54



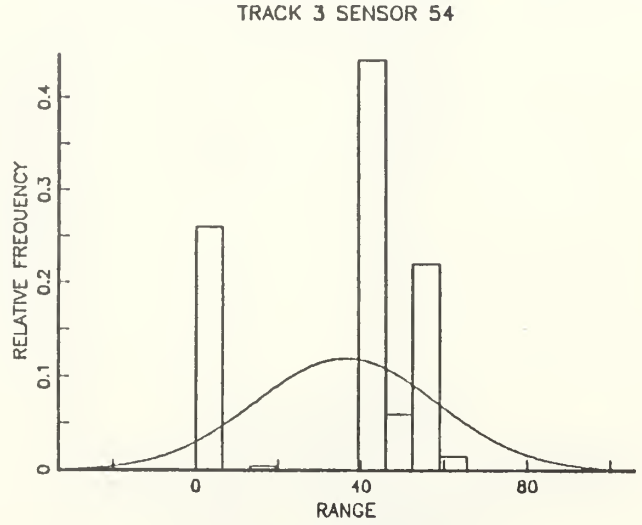
TRACK 2 SENSOR 55

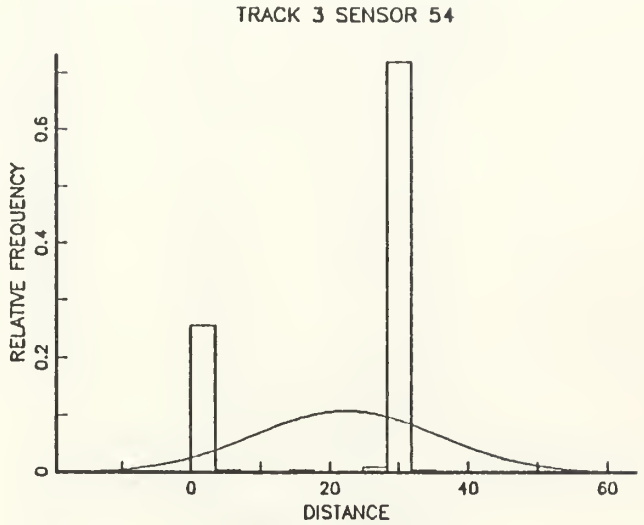


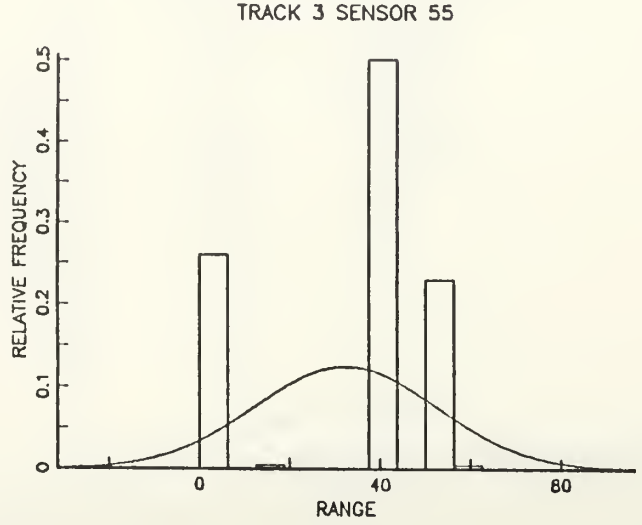
TRACK 2 SENSOR 55

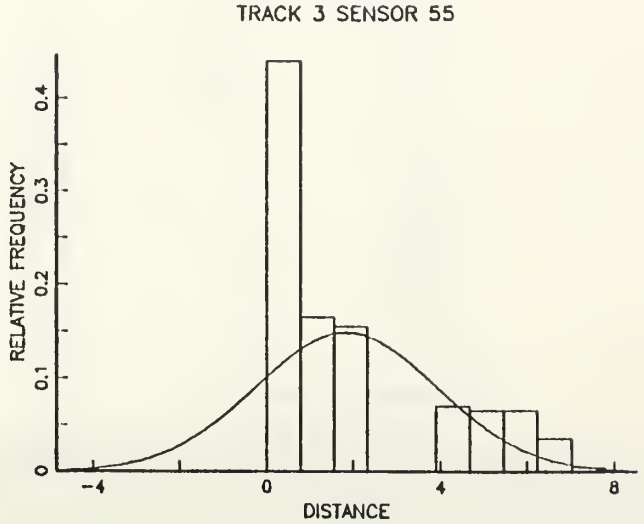


NORMAL VELOCITY ERROR: ST DEV 0.0001

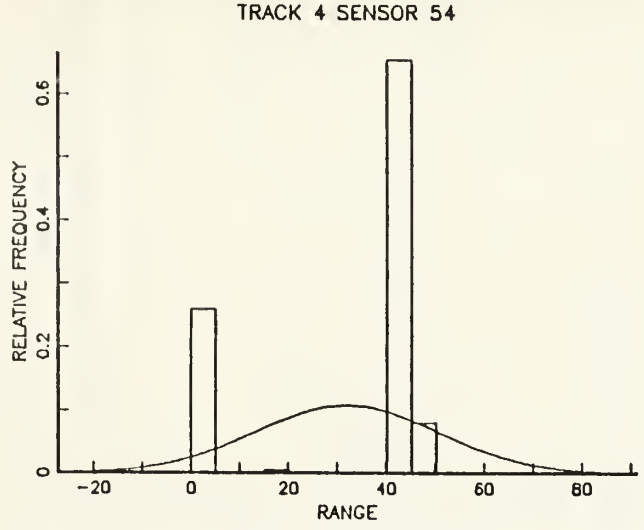


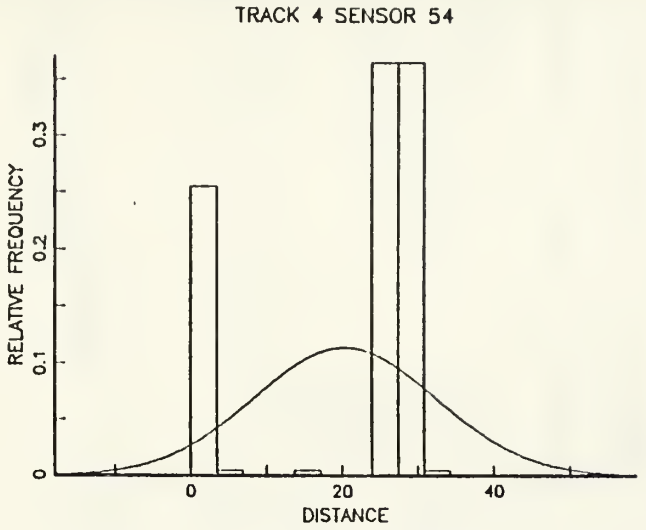


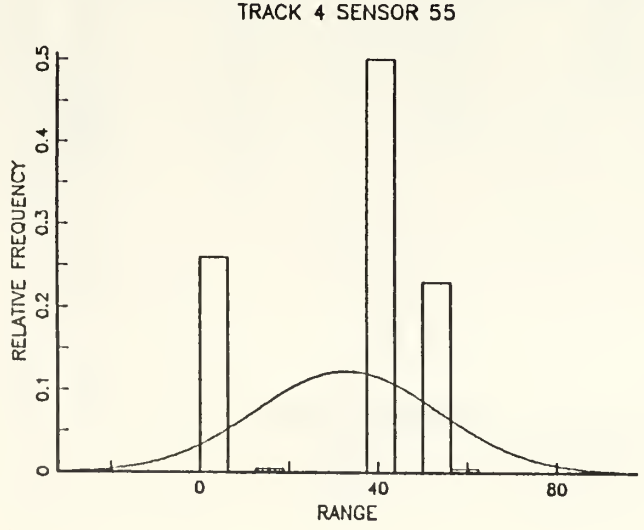


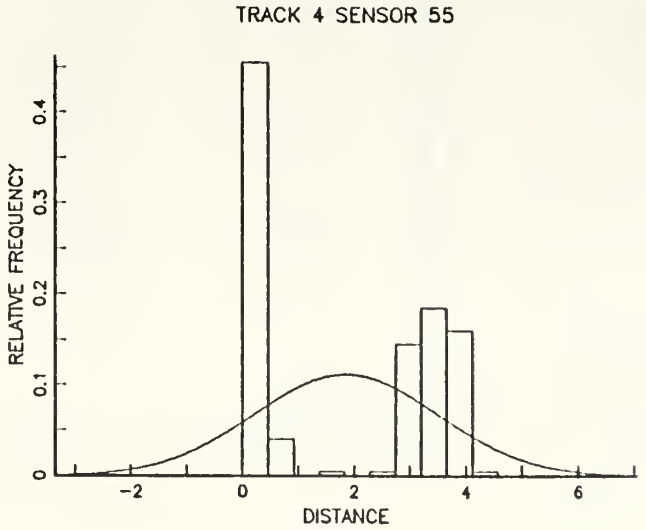


NORMALVELOCITY ERROR: ST DEV 0.0001



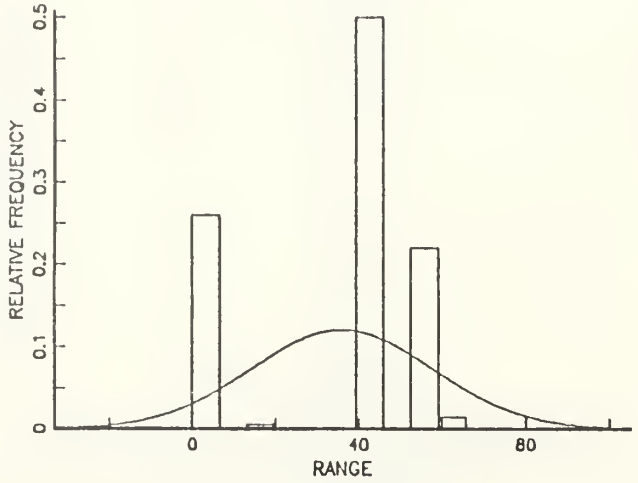




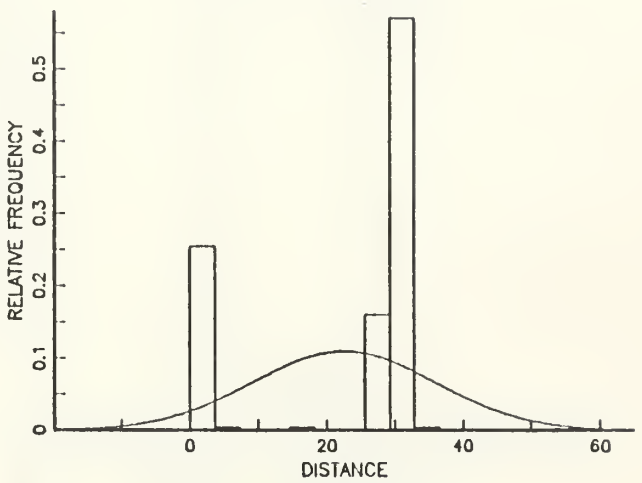


NORMAL VELOCITY ERROR: ST DEV 0.0001

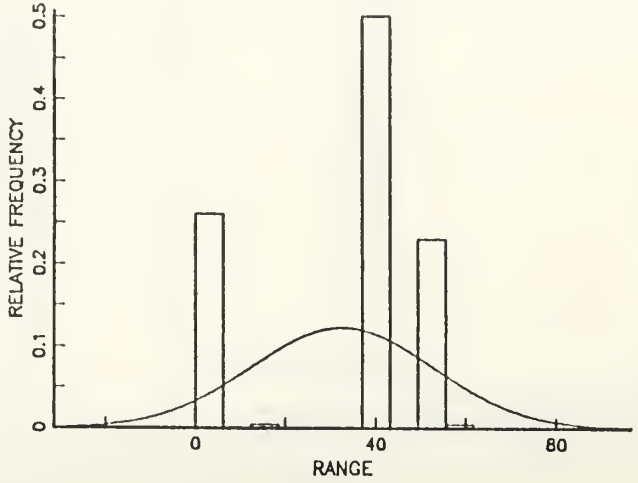
TRACK 5 SENSOR 54



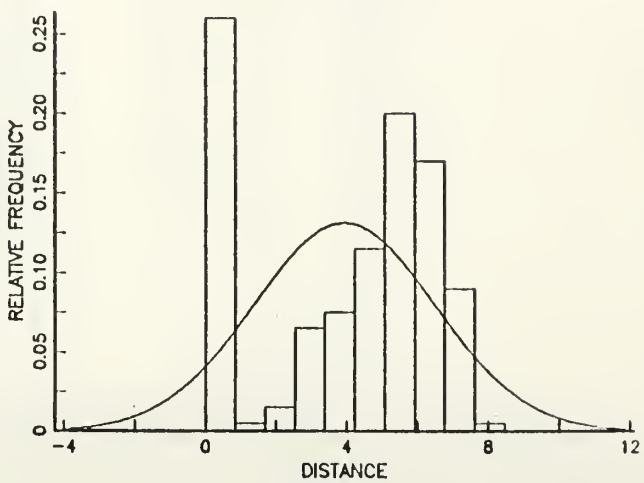
TRACK 5 SENSOR 54



TRACK 5 SENSOR 55

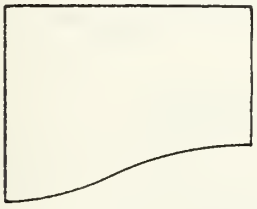


TRACK 5 SENSOR 55



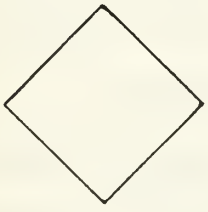
APPENDIX N

This appendix presents a logical flow path for using the various simulation programs presented in this thesis. The following symbol convention is adopted throughout this appendix:



Denotes an output file that is created

Denotes an input from a file



Denotes a decision

The following files are generated or used in the course of operating the simulation:

Global Track Data : This file is the position of the torpedo, at each point count, in global range coordinates of downrange, crossrange, and depth. This file represents the actual position of the torpedo.

Local Track Data: This file is the Global Track Data file converted into local coordinates, relative to a specific sensor on the range.

Sensor Data File: This file refers to the location of each sensor of the range, expressed in range coordinates for downrange, crossrange, and depth.

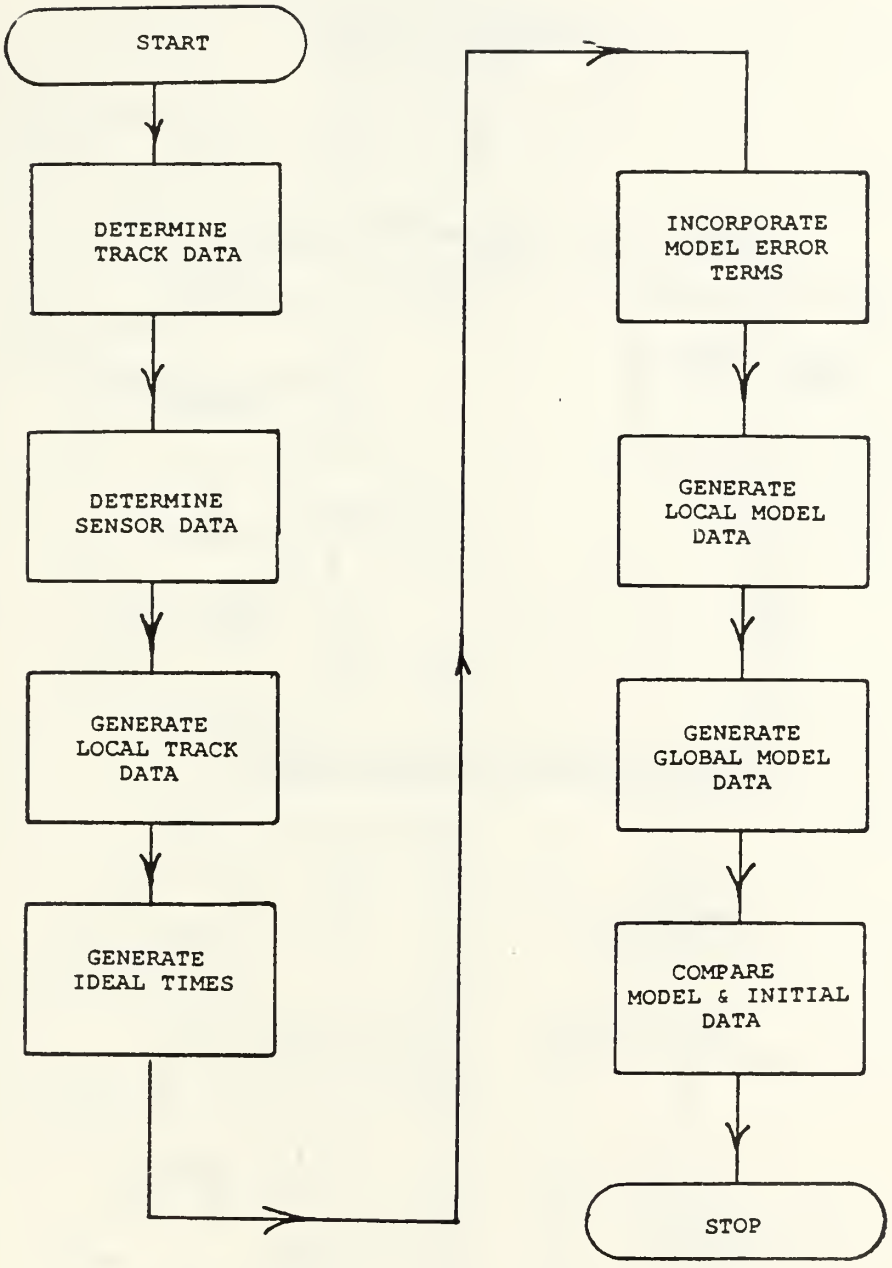
SVP File: This is the file of the sound velocity profile. It represents the speed of sound in water, measured every 25 feet, from the surface downward.

Ideal Times: This is the file of time of flight for sound, from the source to each of the four hydrophones of a single specific sensor, for each point count.

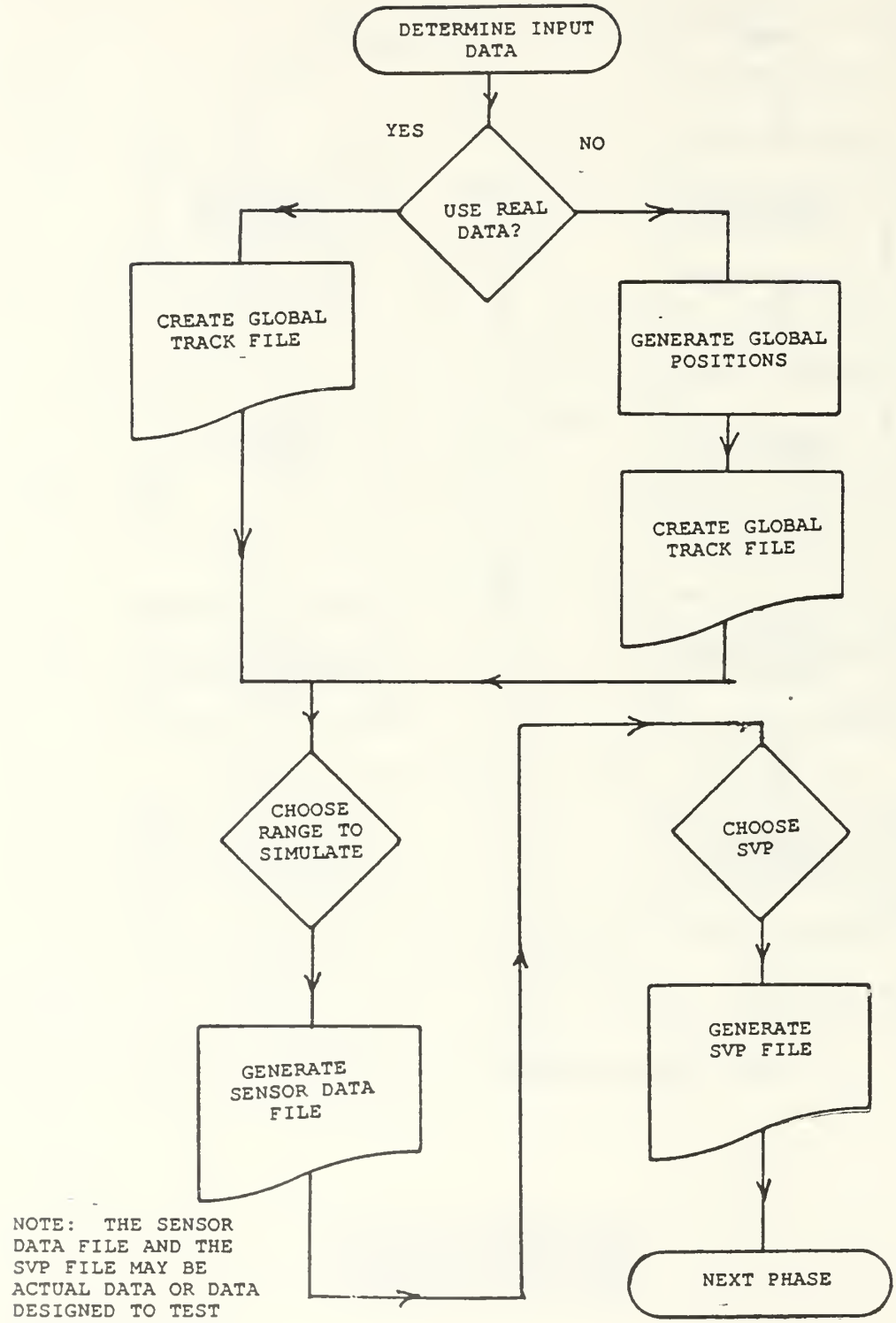
Model Times: The time of flight for sound, as modified by any time variation model being tested. It is the Ideal time file with deliberate error terms introduced.

Local Model Track File: A file of positions in downrange, crossrange, and depth, for the model times, expressed in local coordiantes for a given sensor.

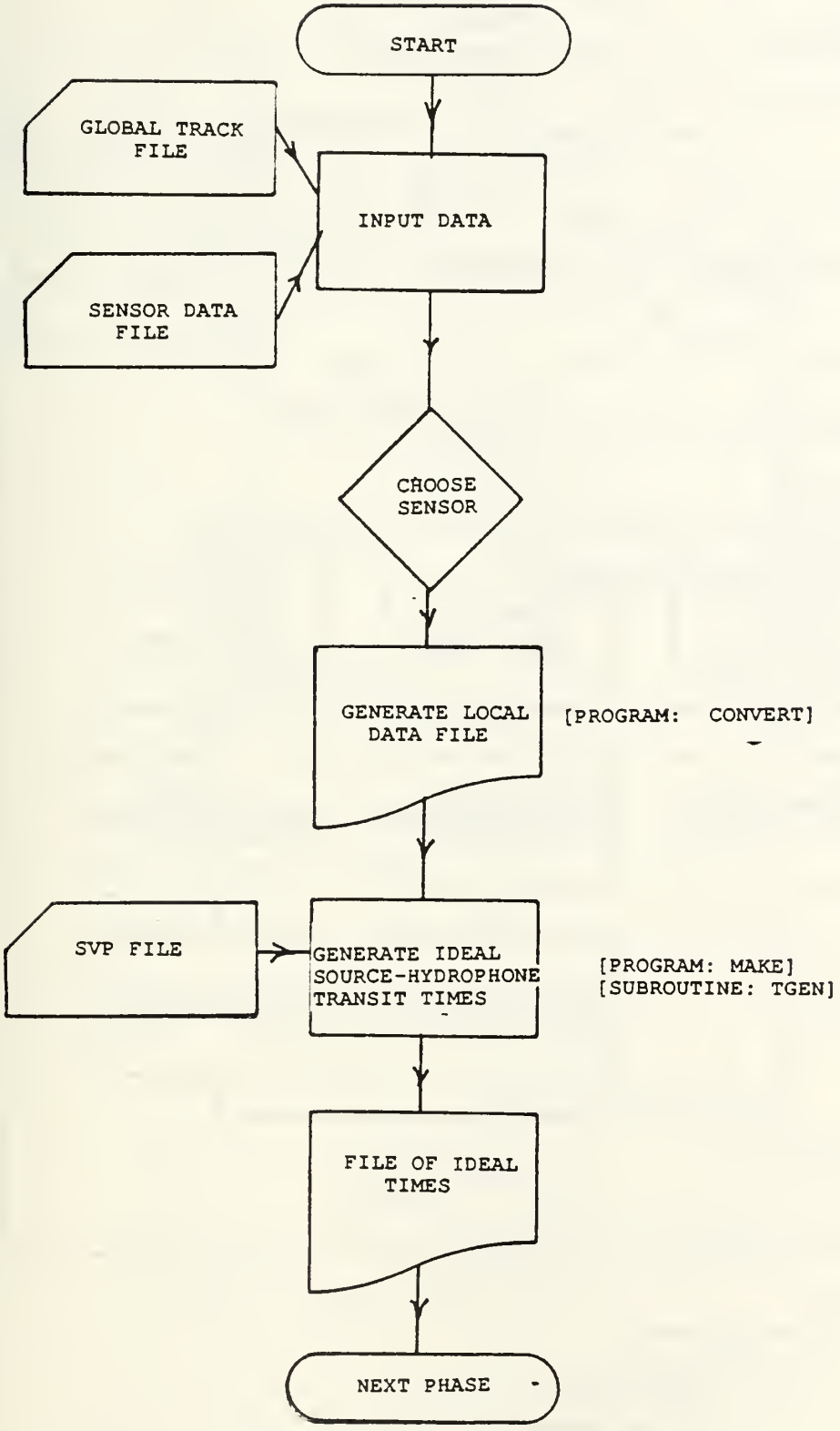
Global Model Track File: The local model track file, converted into global coordinates.

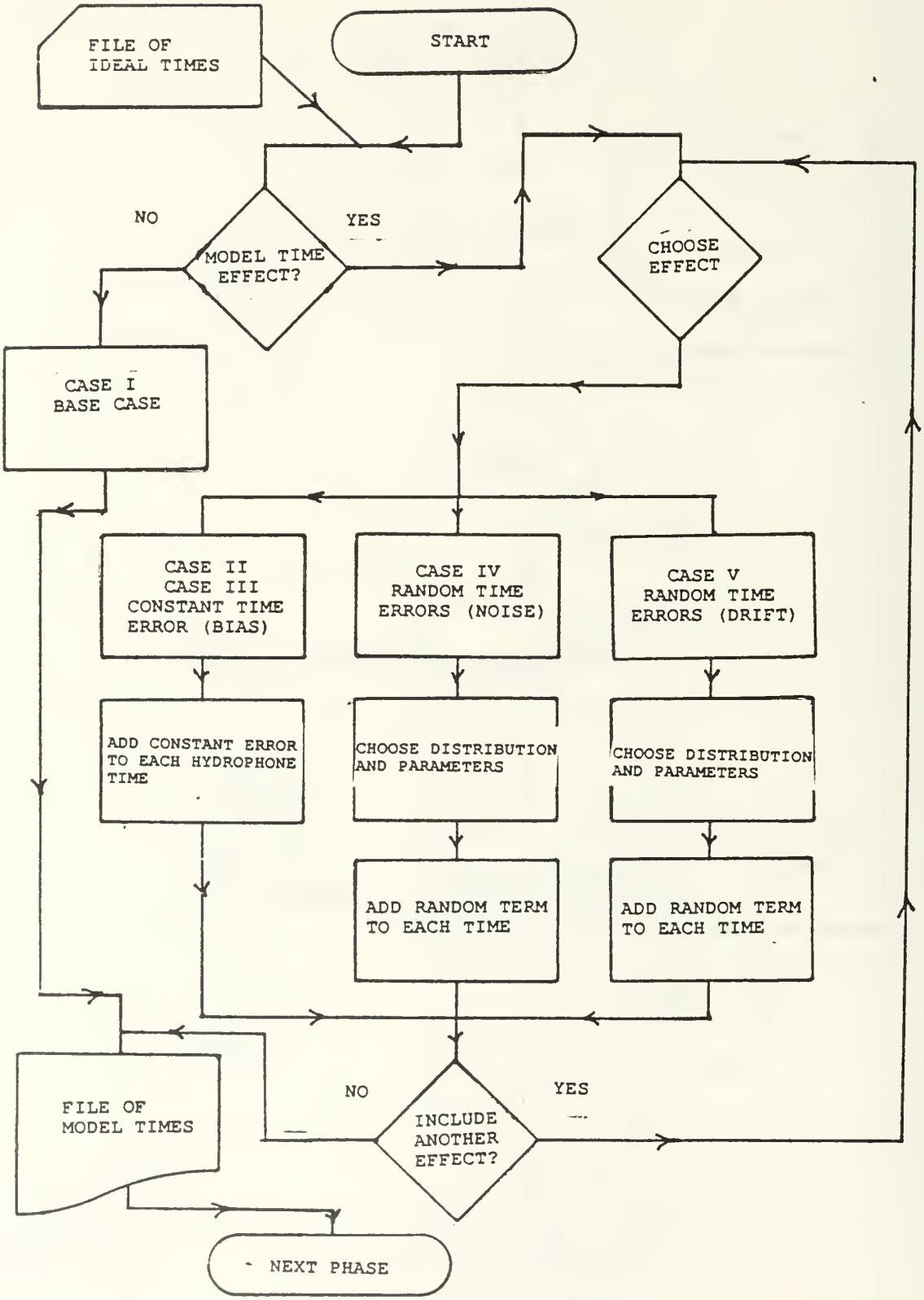


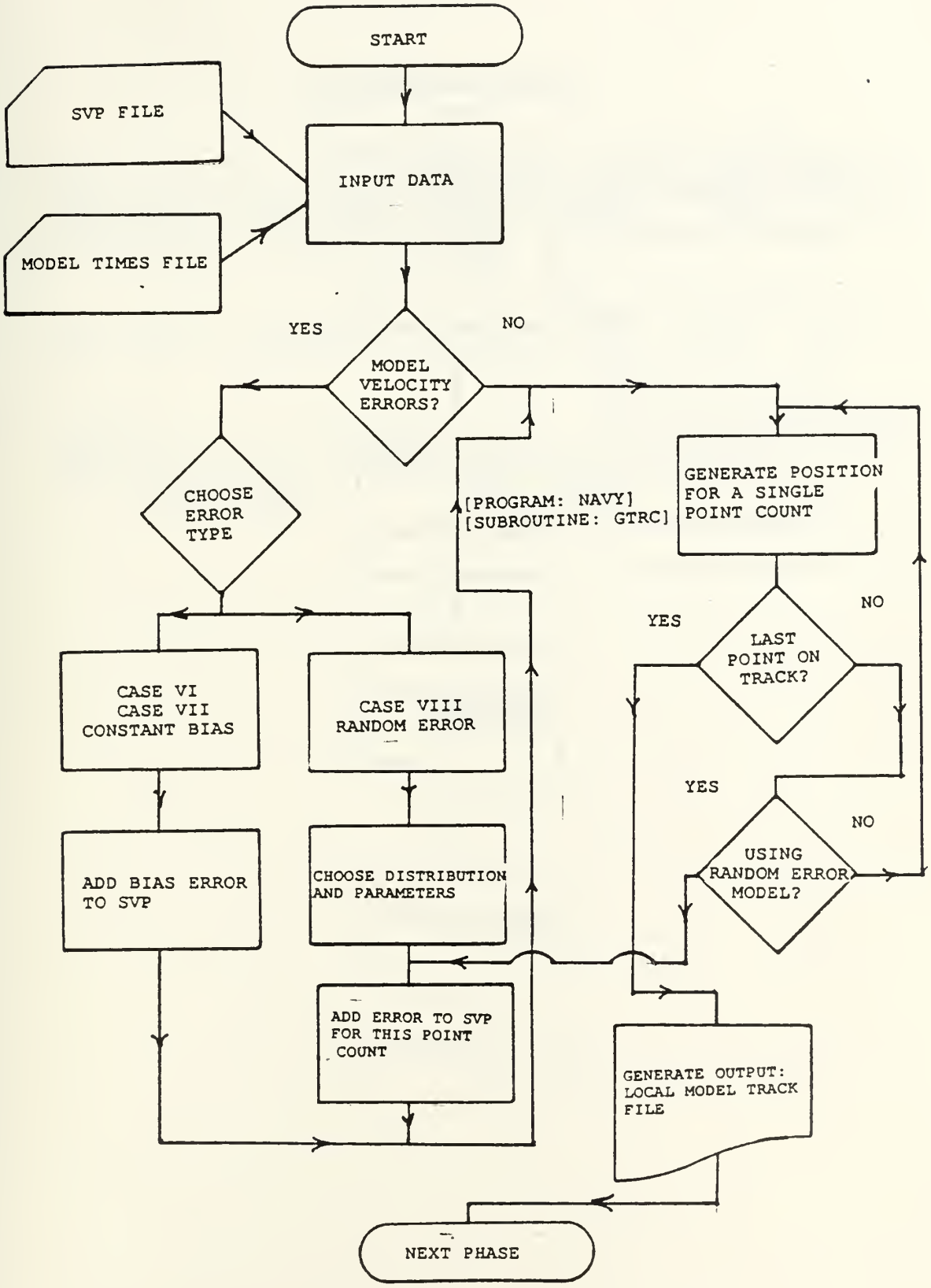
THIS SIMPLIFIED DIAGRAM
IS EXPANDED IN SUBSEQUENT
PAGES OF THIS APPENDIX

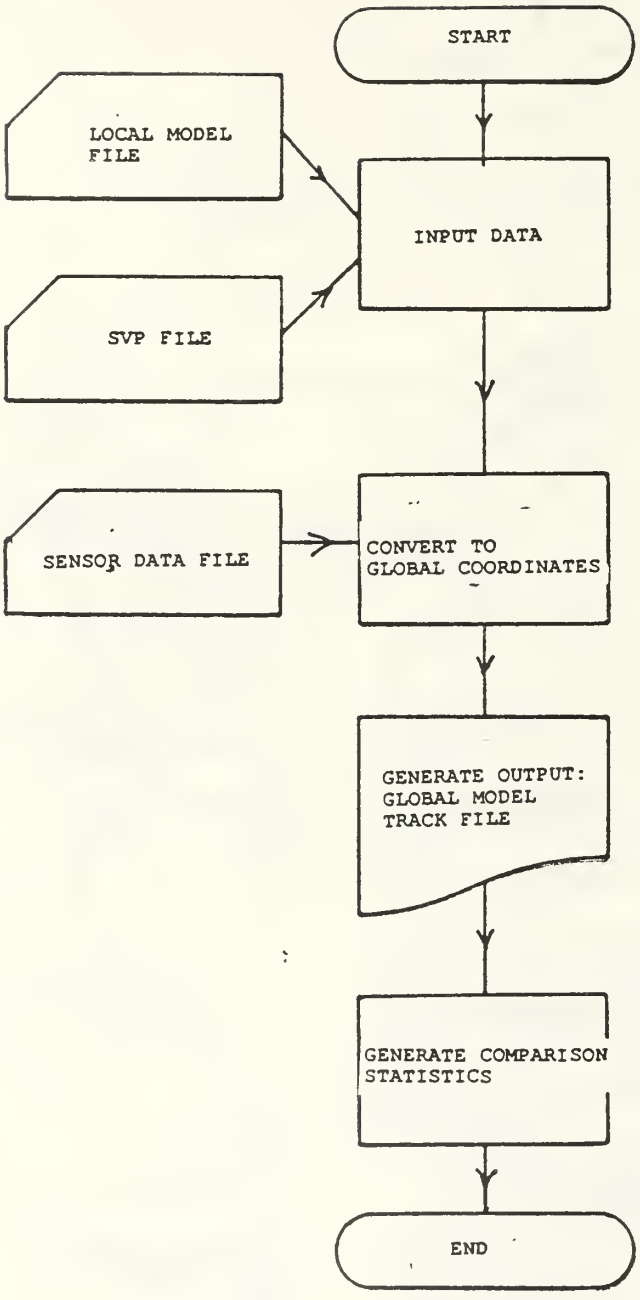


NOTE: THE SENSOR DATA FILE AND THE SVP FILE MAY BE ACTUAL DATA OR DATA DESIGNED TO TEST THE RANGE UNDER CERTAIN CONDITIONS


START
GLOBAL TRACK FILE
INPUT DATA
SENSOR DATA FILE
CHOOSE SENSOR
GENERATE LOCAL DATA FILE
[PROGRAM: CONVERT]
SVP FILE
GENERATE IDEAL SOURCE-HYDROPHONE TRANSIT TIMES
[PROGRAM: MAKE]
[SUBROUTINE: TGEN]
FILE OF IDEAL TIMES
NEXT PHASE


FILE OF IDEAL TIMES
START
NO
MODEL TIME EFFECT?
YES
CHOOSE EFFECT
CASE I BASE CASE
CASE II CASE III CONSTANT TIME ERROR (BIAS)
CASE IV RANDOM TIME ERRORS (NOISE)
CASE V RANDOM TIME ERRORS (DRIFT)
ADD CONSTANT ERROR TO EACH HYDROPHONE TIME
CHOOSE DISTRIBUTION AND PARAMETERS
CHOOSE DISTRIBUTION AND PARAMETERS
ADD RANDOM TERM TO EACH TIME
ADD RANDOM TERM TO EACH TIME
FILE OF MODEL TIMES
NO
INCLUDE ANOTHER EFFECT?
YES
NEXT PHASE


START
SVP FILE
INPUT DATA
MODEL TIMES FILE
YES
NO
MODEL VELOCITY ERRORS?
CHOOSE ERROR TYPE
[PROGRAM: NAVY]
[SUBROUTINE: GTRC]
GENERATE POSITION FOR A SINGLE POINT COUNT
YES
NO
LAST POINT ON TRACK?
CASE VI
CASE VII
CONSTANT BIAS
CASE VIII
RANDOM ERROR
YES
NO
USING RANDOM ERROR MODEL?
ADD BIAS ERROR TO SVP
CHOOSE DISTRIBUTION AND PARAMETERS
ADD ERROR TO SVP FOR THIS POINT COUNT
GENERATE OUTPUT: LOCAL MODEL TRACK FILE
NEXT PHASE


START
LOCAL MODEL FILE
INPUT DATA
SVP FILE
SENSOR DATA FILE
CONVERT TO GLOBAL COORDINATES
[PROGRAM: CONVERT]
GENERATE OUTPUT: GLOBAL MODEL TRACK FILE
GENERATE COMPARISON STATISTICS
END

LIST OF REFERENCES

1. Main, Carl D. Alternative Models for Calculation of Elevation Angles and Ray Transit Times for Ray Tracing Hydrophonic Tracking Data, Master's Thesis, Naval Postgraduate School, Monterey, CA, September, 1984.

2. Camp, Leon Underwater Acoustics, John Wiley and Sons, New York, NY, 1970.

3. Coppens, A. B., Sanders, J. V. , The Eikonal Equation and Ray Theory, Handout for Physics Classes at Naval Postgraduate School, Monterey, CA, 1976

INITIAL DISTRIBUTION LIST

No.		Copies
1.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
2.	Professor R. R. Read, Code 55Re Operations Research Department Naval Postgraduate School Monterey, CA 93943-5000	2
3.	Professor R. N. Forrest, Code 55Fo Operations Research Department Naval Postgraduate School Monterey, CA 93943-5000	2
4.	Commanding Officer Research and Engineering Department Naval Underwater Warfare Engineering Station Keyport, WA 98345	2
5.	Mr. Sam McKeel Code 50 Naval Underwater Warfare Engineering Station Keyport, WA 98345	5
6.	LCDR W. M. Kroshl, USN 4865 Fountain Hall Drive Virginia Beach, VA 93940	5
7.	Terri Turner, Code 30 Naval Postgraduate School Monterey, CA 93943-5000	1
8.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2

Thesis
K8676
c.1

Kroshl
Methodologies for resolving anomalous position information in torpedo range tracking using simulation.

Thesis
K8676
c.1

Kroshl
Methodologies for resolving anomalous position information in torpedo range tracking using simulation.

AMERICAN BINDERY
TOPEKA KANSAS

thesK8676
Methodologies for resolving anomalous po
3 2768 000 78332 8
DUDLEY KNOX LIBRARY